

La rivista dedicata al mondo AMIGA, CDTV e C-64/128

COMMODORE GAZETTE

Audio a 16 bit:

**Clarity 16
alla prova**

Il CeBIT di Hannover:

TUTTE LE NOVITÀ PER AMIGA

Oltre Mandelbrot:

LO SPAZIO DI LYAPUNOV

Prove hardware:

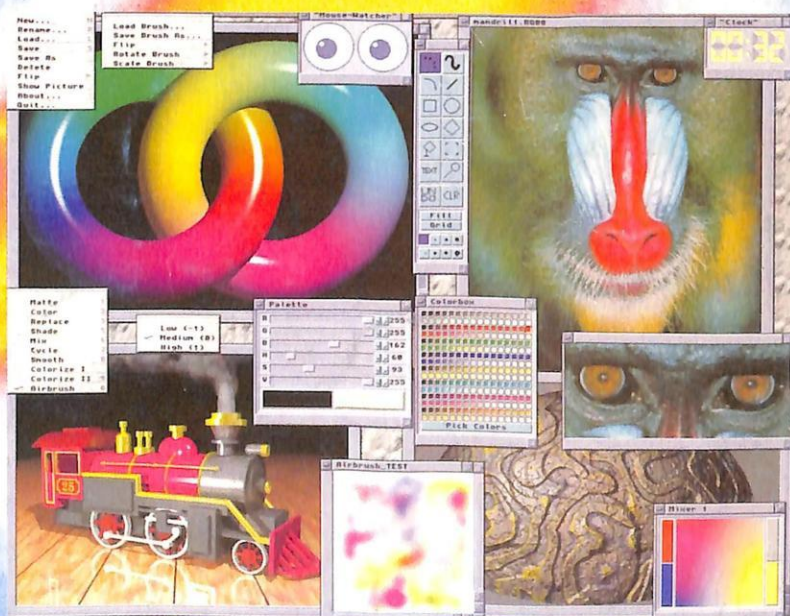
LO SCANNER ALFASCAN PLUS

Amiga 3D:

- ESSENCE DI STEVE WORLEY
- INTERVISTA ALL'AUTORE DI IMAGINE
- TECHNO 3D: LENTI, BRUSH, TEXTURE...
- REAL 3D 2.22

Programmare da zero:

CORSO DI ASSEMBLY



Programmare l'Amiga:

- ESTENSIONI PER AMOS
- CORSO DI MODULA 2
- GESTIONE TESTI IN C

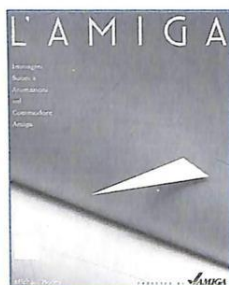
TUTTI I LIBRI IHT

UNA GUIDA DETTAGLIATA PER CONOSCERLI MEGLIO

COLLANA

INFORMATICA

La prima e più prestigiosa collana della IHT Gruppo Editoriale. Guide a sistemi operativi, all'uso del computer e software, alla programmazione; testi che spesso costituiscono la documentazione ufficiale su un prodotto, tutti accuratamente controllati con una completa verifica dei contenuti tecnici.



L'AMIGA: IMMAGINI, SUONI E ANIMAZIONI SUL COMMODORE AMIGA

Un libro molto chiaro che introduce i nuovi utenti di un Amiga a tutte le caratteristiche del loro computer. Gli argomenti trattati sono: l'hardware, la videografia, la generazione di suoni e musica, *Deluxe Music*, *Deluxe Video*, *Deluxe Paint*, l'Amiga BASIC.

416 pagine - L. 60.000 - ISBN 88-7803-000-7



IL MANUALE DELL'AMIGADOS

La documentazione ufficiale realizzata dalla Commodore sul DOS dell'Amiga. Il testo è diviso in tre parti: Il manuale per l'utente, per il programmatore e di riferimento tecnico. Un libro indispensabile sia per i programmatori sia per i neofiti.

376 pagine - L. 60.000 - ISBN 88-7803-002-3



PROGRAMMARE L'AMIGA VOLUME I

Un testo davvero indispensabile per tutti i programmatori in linguaggio C e in linguaggio Assembly. Il libro esamina più di 300 funzioni di sistema dettagliando tutte le strutture disponibili per grafica, animazioni e gestione del multitasking. Non mancano gli esempi.

784 pagine - L. 80.000 - ISBN 88-7803-004-X



PROGRAMMARE L'AMIGA VOLUME 2

La continuazione del testo precedente che tratta in modo approfondito e con chiari schemi la programmazione di tutti i dispositivi di I/O, la generazione di suoni e la sintesi vocale. Un libro che non può assolutamente mancare nella vostra biblioteca tecnica.

528 pagine - L. 70.000 - ISBN 88-7803-005-8



IL MANUALE DELL'HARDWARE DELL'AMIGA

Il testo di riferimento indispensabile per tutti i programmatori che utilizzano il linguaggio Assembly e per i progettisti di hardware per l'Amiga. Il volume è stato scritto dai programmatori della stessa Commodore-Amiga ed è quindi una documentazione ufficiale sull'Amiga.

336 pagine - L. 76.000 - ISBN 88-7803-018-X



FLIGHT SIMULATOR CO-PILOT

Un vero istruttore di volo per tutti coloro che vogliono "volare davvero" con il programma *Flight Simulator* per MS-DOS, C-64, C-128, Apple II, Atari 800 XL e XE. Un bellissimo libro adatto tanto al neofita quanto al pilota già esperto.

152 pagine - L. 30.000 - ISBN 88-7803-001-5



VOLARE CON FLIGHT SIMULATOR

Un vero e proprio corso di volo che propone anche numerose avventure nel cielo ai limiti delle caratteristiche del programma *Flight Simulator* nelle versioni per Amiga, Atari ST e Macintosh. Un libro davvero indispensabile per chi ama i simulatori di volo.

232 pagine - L. 45.000 - ISBN 88-7803-006-6



GUIDA UFFICIALE ALLA PROGRAMMAZIONE DI GEOS

Scritta dagli stessi creatori di *GEOS* per C-64 e C-128, questa guida è indispensabile per conoscere a fondo i segreti di *GEOS* e per sviluppare programmi in standard *GEOS* dotati di finestre, icone, menu, box di dialogo...

592 pagine - L. 60.000 - ISBN 88-7803-003-1

COLLANA CINEMA

Nel cinema arte e tecnologia sono inscindibilmente legate, ed è affascinante scoprire quale intreccio regoli i rapporti tra questi due mondi, apparentemente così diversi. La collana cinema nasce da questo.



LE MILLE LUCI DI HOLLYWOOD

Un libro che vi porta dietro le quinte di film come *Guerre Stellari*, *Star Trek*, *Amadeus*, *Tron*, *E.T.*, *Ritorno al Futuro*, *Apocalypse Now*... e vi svela tutti i segreti di: effetti speciali, computergrafica, fotografia, montaggio, sonoro, scenografia, costumi, trucco, animazioni...

440 pagine - L. 42.000 - ISBN 88-7803-009-0

COLLANA TEMPUS

Un settore ancora tutto da esplorare: quello della ricerca scientifica e tecnologica. La collana offre al pubblico un catalogo quanto più vario possibile che, privilegiando il punto di vista tecnologico, aiuti ad aggiornarsi sul mondo moderno.

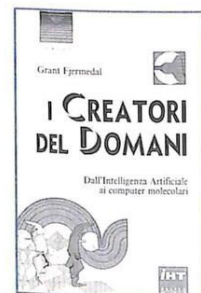


LA MACCHINA E LA MENTE

Alla scoperta dell'Intelligenza Artificiale

Uno dei migliori libri sull'Intelligenza Artificiale oggi disponibili. Douglas Hofstadter, autore di *Gödel, Escher, Bach*, lo ha definito: «Una presentazione ideale dell'IA... vivace e stimolante, scritta con chiarezza, una lettura affascinante».

464 pagine - L. 42.000 - ISBN 88-7803-012-0



I CREATORI DEL DOMANI

Dall'Intelligenza Artificiale ai computer molecolari

Questo testo vi condurrà ai confini del futuro, dove gli scienziati spingono la loro immaginazione ai limiti estremi. Visiterete i principali laboratori di robotica del mondo e scoprirete cosa sono l'esperienza artificiale e il downloading di un cervello...

320 pagine - L. 39.900 - ISBN 88-7803-013-9



COMPUTER IN GUERRA: FUNZIONERANNO?

I rischi e le potenzialità delle nuove tecnologie militari

Nel nostro futuro ci sono guerre stellari, armamenti autonomi e robot killer... A che punto sono i passi in questa direzione? Lo sapevate che più di una volta i computer del NORAD ci hanno fatto rischiare la Terza guerra mondiale?

352 pagine - L. 39.900 - ISBN 88-7803-011-2



INVENTORI DEL NOSTRO TEMPO

Interviste con 16 famosi inventori americani

Un'affascinante raccolta d'interviste a inventori come Wozniak (Apple II), Kurzweil (sintetizzatore musicale), Ted Hoff (microprocessore), Gould (laser), Rosen (satellite geostazionario), Greatbatch (pacemaker impiantabile), Camras (registratore)...

416 pagine - L. 42.000 - ISBN 88-7803-010-4



L'UNIVERSO DEL GIOVEDÌ

Le nuove teorie sull'origine, la natura e il destino dell'universo

Uno dei migliori testi di divulgazione scientifica sulle più recenti teorie riguardanti l'universo. Se volete sapere cos'è stato scoperto negli ultimi 20 anni e quali sono gli interrogativi irrisolti, questo è il libro da leggere.

344 pagine - L. 39.900 - ISBN 88-7803-015-5

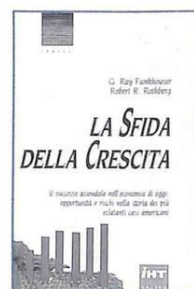


FRONTIERE INVISIBILI

Ingegneria genetica: la sintesi del primo gene umano

Il libro è la storia della competizione tra gli scienziati che hanno creato il primo gene umano (il gene dell'insulina) e che hanno così dato vita all'ingegneria genetica. Il testo è un interessante ritratto della nascita della rivoluzione della biotecnologia.

304 pagine - L. 54.000 - ISBN 88-7803-016-3



LA SFIDA DELLA CRESCITA

Il successo aziendale nell'economia di oggi

Le storie di eccezionali fenomeni di crescita aziendale (IBM, Du Pont, Procter & Gamble, Apple...) e di disastri (Atari, BankAmerica, People Express). Un libro illuminante destinato a dirigenti, imprenditori, investitori, economisti, studenti, docenti...

336 pagine - L. 39.900 - ISBN 88-7803-014-7

COME ACQUISTARE I LIBRI IHT

LIBRERIE

Se la vostra libreria di fiducia ne è sprovvista, potete farveli ordinare specificando il titolo, il codice ISBN e il nostro distributore (RCS Rizzoli Libri - Tel. 02/5095954).

COMPUTERSHOP

I migliori computershop dispongono dei nostri libri.

PER TELEFONO

Potete ordinare telefonando allo 02/794181 - 76022612 - 76022612 - 794122. Riceverete i libri a casa vostra e pagherete al postino.

VIA FAX

Potete inoltrare il vostro ordine allo 02/784021 (24 ore su 24).

VIA POSTA

Potete compilare e spedire il tagliando pubblicato a pagina 95 di questa rivista.

IHT Gruppo Editoriale - Via Monte Napoleone, 9 - 20121 Milano

FLOPPERIA®

S.r.l.

Viale Monte Nero, 15 • 20135 Milano • Tel. (02) 55.18.04.84 r.a. • Fax (02) 55.18.81.05 • MM3 Medaglie D'Oro
Piazza Santa Maria Beltrade, 1 (angolo Via Torino) • 20123 Milano • Tel. (02) 72.00.18.10 • MM1-MM3 Duomo
Flopperia Mail • Vendita per corrispondenza. Importazione diretta. Spedizioni in 24 ore.



COMPUTER STAKAR

386-SX 33 MHz hard disk 120 MB	1.534.000
386-DX 40 MHz hard disk 120 MB	1.959.000
386-DX 40 MHz hard disk 210 MB	2.153.000
386-DX 40 MHz hard disk 340 MB	2.719.000
386-DX 40 MHz hard disk 120 MB	2.317.000
386-DX 40 MHz hard disk 210 MB	2.511.000
386-DX 33 MHz hard disk 120 MB	3.077.000
486-DX 33 MHz hard disk 120 MB	2.911.000
486-DX 33 MHz hard disk 210 MB	3.105.000
486-DX 33 MHz hard disk 340 MB	3.671.000
486-DX 50 MHz hard disk 120 MB	3.181.000
486-DX 50 MHz hard disk 210 MB	3.375.000
486-DX 50 MHz hard disk 340 MB	3.941.000
486-DX 66 MHz hard disk 120 MB	3.502.000
486-DX 66 MHz hard disk 210 MB	3.696.000
486-DX 66 MHz hard disk 340 MB	4.263.000

COMPUTER COMMODORE AMIGA

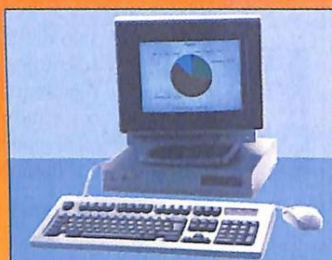
Amiga 500	419.000
Amiga 600	499.000
Amiga 600 HD 40 MB	799.000
Amiga 1200	799.000
Amiga 1200 HD 40 MB	1.099.000
Amiga 1200 HD 80 MB	1.499.000
Amiga 1200 HD 120 MB	1.899.000
Amiga 2000	999.000
Amiga 2000 HD 120 MB	1.699.000
Amiga 4000-30 3-8 hard disk 85 MB	2.595.000
Amiga 4000-30 3-12 hard disk 125 MB	2.795.000
Amiga 4000-40 3-12 hard disk 125 MB	3.995.000
Amiga 4000-40 3-20 hard disk 210 MB	4.295.000

COMPUTER COMMODORE CDTV

CDTV Computer System + Enc. Grolier	1.199.000
Trackball ad infrarossi	179.000
Mouse	115.000
Tastiera	129.000
Genlock PAL	319.000
Scheda Scart	79.000
Caddy x2	39.000
Personal Ram Card 64 KB	195.000
Personal Ram Card 256 KB	575.000

COMPUTER COMMODORE PC

PC 386-SX 25-40 hard disk 40 MB	1.150.000
PC 386-SX 25-80 hard disk 80 MB	1.295.000



PC 386-SX 25-12 hard disk 125 MB	1.395.000
PC 486-SLC 25-40 hard disk 40 MB	1.440.000
PC 486-SLC 25-12 hard disk 125 MB	1.690.000
PC 486-25C 3-12 hard disk 125 MB	1.990.000
PC 486-25C 3-20 hard disk 210 MB	2.350.000
PC 486-33C 3-12 hard disk 125 MB	2.440.000
PC 486-33C 3-20 hard disk 210 MB	2.890.000
PC 486-66C 3-12 hard disk 125 MB	3.620.000
PC 486-66C 3-20 hard disk 210 MB	3.990.000

COMPUTER EUROSYS

Computer forniti con monitor mono 14" e hd 40 MB

386-SX EuroSys 33 MHz	1.390.000
386-DX EuroSys 40 MHz cache 128K	1.530.000
486-DX EuroSys 33 MHz cache 128K	2.140.000
486-DX EuroSys 50 MHz cache 128K	2.890.000
486-DX2 EuroSys 33/66 MHz cache 128K	3.110.000

CONFIGURAZIONI SU MISURA

Prezzi da aggiungere al costo base

con secondo disk drive 3"1/2 da 1.44 MB	95.000
con secondo disk drive 5"1/4 da 1.2 MB	120.000
con scheda SoundBlaster Pro	320.000
con scheda SoundBlaster 16-ASP	525.000
diff. per SuperVGA 1 MB	65.000
diff. per SuperVGA Win. Accel. 16 mil. di colori	110.000
diff. per SuperVGA Local Bus VESA 16 mil.col.	230.000
diff. per SuperVGA S3 Acel. 16 mil.col.	380.000
diff. per mon. col. 14" 1024x768	310.000
diff. per mon. col. 14" 1024x768 Low Emission	510.000
diff. per mon. col. 17" 1280x1024 Low Emission	1.550.000
diff. per mon. col. 20" 1280x1024 Low Emission	2.980.000
con controller IDE cache 2.5 MB Ram (0,5 ms.)	380.000
diff. per hard disk 85 MB IDE 15 ms.	160.000
diff. per hard disk 125 MB IDE 15 ms.	305.000
diff. per hard disk 210 MB IDE 15 ms.	510.000
diff. per hard disk 340 MB IDE 15 ms.	985.000
diff. per hard disk 520 MB IDE 12 ms.	1.880.000
con controller SCSI	110.000
con controller SCSI cache 2 MB Ram (0,4 ms.)	750.000
diff. per hard disk SCSI 210 MB 15 ms.	830.000
diff. per hard disk SCSI 520 MB 12 ms.	1.890.000
diff. per hard disk SCSI 1 GB 11 ms.	2.990.000
con tape streamer 250 MB	730.000
diff. per cabinet minitower 4 posizioni	25.000
diff. per cabinet tower 5 posizioni	140.000
diff. per 2 MB Ram (solo per 386 e 386-SX)	65.000
diff. per 4 MB Ram (tutti)	215.000
diff. per 8 MB Ram (solo per 386 e 486)	510.000



STAMPANTI

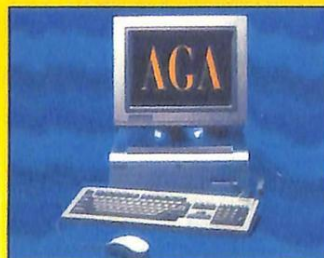
Seikoshia SL-90	475.000
Seikoshia SL-95 color	625.000
Star LC-20	395.000
Star LC-24/20	650.000
Star LC-100 color	450.000
Star LC-24/100 24 aghi	595.000
Star SJ-48 InkJet	710.000
Star LC-24/200	820.000
Star LC-24/200 color	880.000
Commodore MPS 1230	315.000
Commodore MPS 1270	299.000
Commodore MPS 1550 color	415.000
Nec P20	699.000
Nec P30 136 colonne	899.000
Nec P60	1.190.000
Nec P70 136 colonne	1.440.000
Canon BJ-10/EX	660.000
Canon BJ-20	795.000
HP DeskJet 500	875.000
HP DeskJet 500-C Color	1.160.000
HP DeskJet 550-C Color	1.625.000
HP LaserJet III-P	3.082.000
HP LaserJet 4	4.105.000
HP LaserJet 4-M	5.414.000
MicroLaser Plus 9 pag/min 0,5 MB Ram	1.699.000
MicroLaser Plus PS 9 pag/min 1,5 MB Ram	2.450.000
MicroLaser XL 16 pag/min 0,5 MB Ram	4.350.000
MicroLaser XL PS 16 pag/min 1,5 MB Ram	5.090.000
Epson LX-400 80 col. 180 cps 9 aghi	399.000
Epson LQ-100 80 col. 180 cps 24 aghi	539.000
Epson Stylus 800	819.000
Epson LX-850 80 col. 200 cps 9 aghi	640.000
Epson LX-1050 136 col. 200 cps 9 aghi	910.000
Epson FX-850 80 col. 260 cps 9 aghi	860.000
Epson FX-1050 136 col. 260 cps 9 aghi	1.050.000
Epson LQ-450 80 col. 220 cps 24 aghi	795.000
Epson LQ-570 80 col. 250 cps 24 aghi	840.000
Epson LQ-1070 136 col. 250 cps 24 aghi	1.190.000
Epson LQ-870 80 col. 330 cps 24 aghi	1.250.000
Epson LQ-1170 136 col. 330 cps 24 aghi	1.580.000
Epson LQ-1060 col. 136 col. 300 cps 24 aghi	1.880.000
Epson LQ-2550 col. 136 col. 600 cps 24 aghi	2.460.000
Epson EPJ-200 inkjet 136 col. 1 pag/min	2.250.000
Epson SQ-870 inkjet 80 col. 550 cps	1.410.000
Epson SQ-1170 inkjet 136 col. 550 cps	2.050.000
EPL-4000 laser 300 dpi 6 pag/min 0,5 MB	1.630.000
EPL-4100 laser 300/600 dpi 6 pag/min 0,5 MB	1.940.000
EPL-4300 laser 300/600 dpi 6 pag/min 1 MB	2.060.000
EPL-7500 laser 300 dpi 6 pag/min PS 2 MB	4.130.000



MULTIMEDIA

SOUNDMACHINE	259.000
Soundblaster 2.0 + joy + casse + 3 giochi	
SOUNDBLASTER PRO DELUXE	325.000
La più venduta scheda audio musicale con bus ISA.	
SOUNDBLASTER PRO BASE MCA	549.000
Scheda audio musicale con bus Microchannel (PS2).	
SOUNDBLASTER PRO 16 ASP	539.000
Nuova scheda con frequenza di campionamento da 5 a 44.1 kHz a 16 bit.	
WAVEBLASTER	449.000
Trasforma la Soundblaster in un sintetizzatore high-quality. Con software e drivers per Windows 3.1.	
MIDIBLASTER	485.000
Modulo musicale Midi high-quality. Con software, clips musicali, cavi e alimentatore.	
SOUNDBLASTER MIDI KIT	99.000
Adattatore Midi per Soundblaster + software.	
SOUNDBLASTER CD-ROM	649.000
Lettore interno CD-Rom per Soundblaster con 2 CD.	
SOUNDBLASTER MULTIMEDIA KIT	999.000
SB MULTIMEDIA KIT EDU-TAINMENT	1.095.000
SB MULTIMEDIA CREATIVE KIT	1.195.000
Kit Multimediale con lettore CD-ROM, scheda Soundblaster, casse stereo, ecc. Varie configurazioni.	
VIDEOBLASTER	649.000
Scheda digitalizzatrice per PC VGA con driver per Windows 3.1.	
VIDEOBLASTER DEVELOPER KIT	159.900
Set aggiuntivo per sviluppare programmi utilizzando la scheda Videoblaster.	
VIDI MEDIA PRO PLUS	949.000
Scheda digitalizzatrice PC con ingressi VC e S-VHS supporta anche S-VGA 1024x768, software anche per Windows 3.1.	

Disponibili inoltre centinaia di altri articoli hardware e software pronta consegna. Richiedete il nostro catalogo gratuito



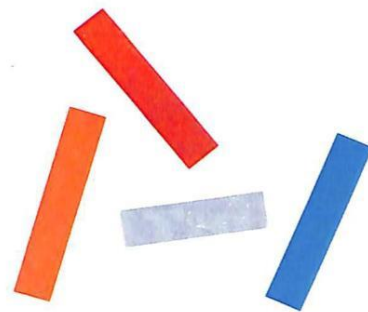
Orari: 10-13, 15.30-19.30. Chiuso Lunedì mattina.

Tutti i prezzi sono comprensivi di I.V.A.



Disponibili tutte le ultime novità per PC, Amiga e console. Per ordinazioni:
Tel. 02/5518.0484-02/7200.1810 • Fax (24h) 02/5518.8105

SOMMARIO



In copertina: un'immagine dello spazio di Lyapunov (pag. 56) e l'interfaccia della scheda Rainbow 3 (pag. 20)

ARTICOLI

- 18 TUTTE LE NOVITÀ DEL CEBIT '93**
Dai nuovi PC Commodore all'A4000 Tower, nuove soluzioni grafiche per Amiga, DCC della Philips contro MD della Sony...
- 24 CLARITY 16: LE NUOVE FRONTIERE AUDIO DELL'AMIGA**
La prova hardware del primo campionatore a 16 bit a basso costo
- 30 AMIGA 3D**
Arrivano nuove essenze per Imagine
Bit Movie Art
Intervista a Mike Halvorson
Real 3D 2.22
Dietro all'immagine
News 3D: Impulse Graphics World
Techno 3D: Animazioni con fotogrammi ripetuti, le lenti con Real 3D, brush minuti con Pixel 3D, tutte le texture che desiderate, messaggi nascosti
Posta 3D: Una grande passione mi brucia il cor, un lungometraggio cyberpunk
- 46 ALFASCAN: SCANNER A TONALITÀ DI GRIGI**
La prova hardware di un package completo che comprende anche un programma di OCR
- 51 LE APPLICAZIONI SCIENTIFICHE SULL'AMIGA**
I vantaggi e gli svantaggi dell'uso dell'Amiga, e le librerie matematiche
- 56 LO SPAZIO DI LYAPUNOV OVVERO LA BELLEZZA DEL CAOS**
Esploriamo con il nostro Amiga un insieme matematico affascinante e complesso
- 59 QUI NEW YORK, STATI UNITI**
Tutto sul CD-ROM per Amiga, Art Expression e TypeSmith della SoftLogik, alla scoperta del DCTV, Deluxe Paint IV AGA...
- 64 LA GESTIONE DEI TESTI: ANALISI DELLE FUNZIONI**
Continua la descrizione del modulo GestioneTesti.c
- 68 LE STRUTTURE DI DATI E L'INPUT/OUTPUT**
Corso di Assembly da zero sull'Amiga: ottava puntata
- 74 SPAZIO AMOS**
Le estensioni AMOS al microscopio
- 78 CORSO DI PROGRAMMAZIONE IN MODULA 2**
Ultima puntata: l'organizzazione e la gestione delle strutture Grafo

RUBRICHE

- 4 NOTE EDITORIALI**
La parola al direttore
- 6 LA POSTA DELLA GAZETTE**
La voce dei nostri lettori
- 9 SOFTWARE GALLERY**
Art Nouveau
B-17 Flying Fortress
Superfrog
Entity
Abandoned Places 2
Prime Mover
- 12 WORLD NEWS**
Novità sull'Amiga da tutto il mondo
- 14 PRODUCTIVITY UPDATE**
Le novità del software di utility
- 16 LUDO NEWS**
Le novità dell'ECTS di Londra
- 86 COMPUTER E DIDATTICA**
Intervistiamo il BTD
- 90 COMPUTER NEWS**
Novità dall'Italia e dall'estero
- 91 PAGINE GIALLE**
Dove acquistare il vostro hardware e software, dove far riparare il vostro computer
- 92 CLASSIFIED**
Piccola pubblicità dei nostri lettori
- 95 SERVIZIO LETTORI**
Tagliandi per Classified, e per ordini di libri e videocassette

APRILE/MAGGIO 1993
Anno VIII, N.3





Direttore responsabile: Massimiliano M. Lisa
Redazione: Nicolò Fontana-Rava, Giovanni Varia
Collaborazione editoriale: Avelino De Sabbata, Fulvio Piccoli, Alfredo Prochet, Alfredo Distefano, Antonio De Lorenzo, Paolo Cardillo, Stefano Franzato, Davide Marazza, Giovanni Zito, Marco Dufour, Paolo Germano, Marco Milano, Luca Favenzi
Corrispondenti USA: William S. Freilich, Daniela D. Freilich
Collaborazione editoriale USA: Matthew Leeds, Eugene P. Mortimore, Morton A. Kevelson
Segretaria di redazione: Silvia Alberti
Impaginazione e grafica: Andrea De Michelis
Fotografie: A.&D.
Disegni: M.P., G. F.

Direzione, Redazione, Amministrazione: IHT Gruppo Editoriale S.r.l. - Via Monte Napoleone, 9 - 20121 Milano

Fotocomposizione: IHT Gruppo Editoriale S.r.l. - Divisione grafica

Fotolito: Colour Separation Trust S.r.l. - Via Melchiorre Gioia, 61 - 20124 Milano

Stampa: Amicare Pizzi S.p.A. - Via A. Pizzi, 14 - 20092 Cinisello Balsamo (MI)

Distribuzione per l'Italia: Messaggerie Periodici S.p.A. - V.le Famagosta, 75 - 20142 Milano - Tel. 02/895921 - aderente A.D.N.

Pubblicità: IHT Gruppo Editoriale S.r.l. - Via Monte Napoleone, 9 - 20121 Milano - Tel. 02/794181-799492-76022612-794122 - Fax 02/784021 - Telex 334261 IHT I

Abbonamenti: IHT Gruppo Editoriale - Servizio Abbonati - Via Monte Napoleone, 9 - 20121 Milano. Linee per registrazione e informazioni sugli abbonamenti: 02/794181 - 799492 - 76022612 - 794122

Costo abbonamenti: Italia 6 numeri L. 48.000 - 12 numeri L. 96.000 - 24 numeri L. 192.000 - 36 numeri L. 288.000

Estero: Europa L. 150.000 (10 numeri), Americhe, Asia... L. 200.000 (10 numeri). Per abbonarsi è necessario inviare una lettera di richiesta a: IHT Gruppo Editoriale S.r.l. - Via Monte Napoleone, 9 - 20121 Milano unendo un assegno bancario o un vaglia postale

Arretrati: Ogni numero arretrato: L. 16.000 (spedizione compresa)

Autorizzazione alla pubblicazione: Tribunale di Milano n. 623 del 21/12/85. Periodico mensile. Sped. in abb. post. gr. III/70. ISSN: 0394-6991 La IHT Gruppo Editoriale è iscritta nel Registro Nazionale della Stampa al n. 2148 vol. 22 foglio 377 in data 5/6/1987

Commodore Gazette è una pubblicazione IHT Gruppo Editoriale. Copyright © 1992 by IHT Gruppo Editoriale S.r.l. Tutti i diritti riservati. Nessuna parte della rivista può essere in alcun modo riprodotta senza autorizzazione scritta della IHT Gruppo Editoriale. Manoscritti e foto originali, anche se non pubblicati, non si restituiscono. I contributi editoriali (di qualunque forma), anche se non utilizzati, non si restituiscono. Non si assume alcuna responsabilità per eventuali errori od omissioni di qualsiasi tipo. Commodore Gazette è un periodico indipendente non connesso in alcun modo con la Commodore Business Machines Inc. né con la Commodore Italiana S.p.A. PET, CBM, Vic-20, C-64, C-128, Amiga, CDTV... sono marchi protetti della Commodore Business Machines. Commodore è un marchio di proprietà riservata della Commodore Italiana S.p.A. Nomi e marchi protetti sono citati senza indicare i relativi brevetti.



Associato alla U.S.P.I. (Unione Stampa Periodica Italiana)

NOTE EDITORIALI

LA PAROLA AL DIRETTORE

Egregio Signor Massimiliano Lisa, direttore del mensile *Commodore Gazette*, nonché autore, regista e conduttore del settimanale televisivo *Informatica VideoMagazine*, le scrivo per manifestarle il mio disappunto sull'impostazione con il quale è stato realizzato il programma da lei ideato. In quanto lettore di *Commodore Gazette* e possessore di un sistema A2000, auspicavo a uno spazio dedicato all'Amiga all'interno della sua trasmissione. Il fatto stesso che lei ne fosse l'ideatore e il regista, oltre che il conduttore, mi hanno fatto sperare in una possibile partecipazione e rivalutazione di una macchina come l'Amiga, ma alla fine della prima e seconda puntata di *IVM* la mia delusione è stata enorme. Non riesco a capire come un programma che si consideri dedicato all'informatica in generale possa tralasciare le macchine della Commodore. Eppure ci sarebbe da parlare, basterebbe invitare Antonio De Lorenzo... e si potrebbe parlare di videografica per ore (quella alla portata di tutti noi, e non quella realizzata su stazioni Silicon Graphics)... E invece, nella sua trasmissione c'è una forma di oscuro pseudo-razzismo nei confronti delle nostre macchine. Un esempio evidente è la puntata dedicata a CD-I... sistemi multimediali con esperti di IBM, Kodak e Philips e nella quale non si è nemmeno lontanamente accennato al CDTV...

Vincenzo Morra, Napoli

Gentile signor Morra, le rispondo da questa rubrica, perché è da questa stessa pagina che nel numero 2/93 ho invitato i nostri lettori a seguire *IVM*. Innanzitutto, mi spiace che lei abbia avuto questa impressione di black-out informativo nei confronti dell'Amiga, ma le cose vanno viste in una prospettiva diversa dalla sua. Che in *IVM* ci sia una predominanza di rubriche dedicate ai PC è vero, ma è anche una scelta obbligata. Una trasmissione televisiva ha dei costi di produzione e degli obiettivi di audience che non sono paragonabili a quelli di *Commodore Gazette*. Se per la IHT è possibile pubblicare una rivista su carta dedicata ai computer Commodore e tenerla in vita con le semplici vendite in edicola e con un numero d'inserzionisti limitato (provi a confrontare le pagine di pubblicità presenti su questa rivista con quelle di una per PC), lo stesso non si può dire per un settimanale televisivo. Esiste un preciso rapporto tra la vastità di un mercato, la presenza d'inserzionisti e la

fattibilità di un progetto. Prova ne è che tutti gli abituali inserzionisti di *Commodore Gazette* sono stati contattati per *IVM*, ma nessuno ha deciso d'investire nella trasmissione, e lo stesso vale anche per la *Commodore Italiana*. Dal momento che una trasmissione televisiva si finanzia con gli sponsor e gli spot, e dal momento che questi attualmente sono tutti legati al mondo PC, la trasmissione non può quindi che avere questa impostazione. Non esiste però alcuna forma di ostracismo. Non appena la *Commodore* presenterà novità di rilievo (nuovi Amiga, CDTV 2...) non mancheremo di parlarne con lo spazio che meritano.

Per dimostrarle poi come ogni cosa abbia una sua motivazione precisa, rispondo anche alle sue considerazioni specifiche. Nella terza puntata abbiamo parlato del *Bit Movie '93*. Bene, proprio in quell'occasione avevamo invitato a partecipare sia Antonio De Lorenzo, sia rappresentanti del *Bit Movie*, ma nessuno ha potuto intervenire. Per quel che riguarda il discorso sul CD-I, nella seconda puntata l'oggetto della nostra attenzione non era il CD-I in sé, ma la presentazione della scheda MPEG che consente di avere film su CD, argomento che ha consentito di dare spunto al dibattito che è poi seguito. In verità, avevamo contattato anche la *Commodore*, ma mentre la Philips è stata in grado di presentare con la scheda MPEG in prototipo, la *Commodore* non era in grado di fare altrettanto. Pertanto, la scelta è caduta obbligatoriamente sulla Philips, la quale ha espresso il desiderio che al dibattito successivo non fosse presente la *Commodore* perché ne sarebbe nato un confronto/scontro tra CDTV e CD-I che non desideravano avvenisse.

Spero di aver chiarito a lei e a tutti i nostri lettori la mia posizione nei confronti della vicenda. Il fatto che io diriga da sette anni questa rivista credo sia sufficiente prova del mio coinvolgimento nei confronti dei prodotti Commodore. Purtroppo, ad oggi non ci sono le condizioni di mercato tali da rendere possibile una maggiore attenzione di *IVM* nei confronti dell'Amiga. Sono io il primo a sperare che questo cambi al più presto. Ritengo comunque ugualmente interessante per tutti seguire una trasmissione come *IVM*. Anche se si è possessori esclusivamente di un Amiga confrontare la propria macchina con altre realtà non può che far bene. Stesso dicasi per la computergrafica: se non ci si ferma a quella Amiga e si fanno dei paragoni si possono anche trarre nuovi spunti per i propri progetti.

M.L.

6-10 MAGGIO 1993 - QUARTIERE FIERA

**È TEMPO DI
“METTERE A BUDGET”**



**MOSTRA MERCATO
DELL'INFORMATICA E DELLA
TELEMATICA PER LO STUDIO,
L'HOBBY, LA CASA**

INGRESSO DALLE 9.00 ALLE 18.00 VIA SPINOLA, (PORTA MECCANICA) M1-AMENDOLA

Segreteria Operativa

Fiera Milano Infotelemarket S.r.l.

Largo Domodossola, 1 — 20145 Milano

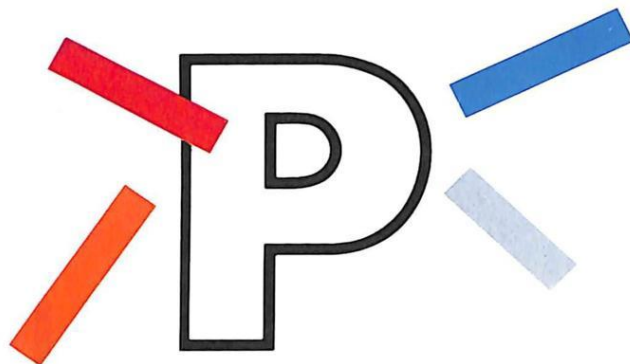
Ingresso Uffici da: Porta Meccanica — Padiglione 20

Tel. (02) 48008619/48 — Fax: (02) 48008571



LA POSTA DELLA GAZETTE

LA VOCE DEI NOSTRI LETTORI



IL MONITOR 1950 È IRREPARABILE?

Vorrei segnalare tramite la rubrica "la voce dei nostri lettori", un episodio negativo tuttora in corso. Ho acquistato a marzo del '91 un monitor Commodore 1950 coperto da regolare garanzia della Commodore Italiana per il mio A2000. A maggio del '92, all'accensione il suddetto non dava segni di vita. Portato all'assistenza Commodore di Genova, mi sento dire che è IRREPARABILE, perché dovrebbero perdersi dietro troppo tempo e che la Commodore (di Milano) si limita a sostituire (o riparare) solo quelli ancora in garanzia!

Ho tentato presso altri laboratori, con esiti sino a oggi negativi, siccome sembra che i componenti guasti siano irreperibili. Ottocentotrentamila lire gettate (+ scheda deinterlacciante). Splendida politica commerciale, questa adottata dalla Commodore, tesa sicuramente a incrementare la vendita dei suoi prodotti, a utenti che già senza episodi come il mio, sembra che lentamente, ma inesorabilmente, stiano spostando le loro attenzioni verso altri lidi. P.S.: Al centro di assistenza di Genova mi era stato proposto, per venirmi incontro, il ritiro del mio 1950 in cambio di un VGA di non si sa quale marchio coreano, versando una somma di circa 600 mila lire.

Alberto Giuseppe Devoto
Corasco

CD-ROM PER AMIGA

Vi scrivo per chiedervi un consiglio: ho un A500 più l'HD A590, 1 MB di chip più 2 MB di fast, Kick 1.3.

Da qualche tempo sono interessato a un drive per CD-ROM, non potendo usare l'A570 contemporaneamente all'Hard Disk, e che il CDTV è sostanzialmente un altro A500 identico a quello che ho già (Kick 1.3!). La migliore soluzione potrebbe essere un lettore via SCSI, sì ma quale? Va bene anche uno per PC o per Apple?

Nelle pagine del numero 8/92, nell'articolo con le "Ultime notizie dagli USA", ho letto che la ditta Xetec produce del software compatibile con tutti i drive, vorrei sapere se questa è l'unica soluzione, o se ce ne sono altre

migliori. In definitiva, vorrei sapere, che cosa bisogna procurarsi per poter usare un lettore CD-ROM su Amiga, credo che questo interesserebbe pure i possessori di A2000, A3000 e A4000. Non capisco come la Commodore, o un'altra azienda, non abbia ancora prodotto un drive appositamente per Amiga.

Luca Girardi
Venezia

Per approfondimenti sul tema del CD-ROM per Amiga, la rimandiamo all'articolo "Qui New York, Stati Uniti", pubblicato a pagina 59 di questo stesso numero.

Dal momento che però abbiamo ricevuto diverse lettere sul tema CD-ROM, cogliamo l'occasione per puntualizzare alcune cose. Innanzitutto, va detto che la Commodore ha annunciato la futura disponibilità di lettori di CD-ROM per la linea Amiga. Quindi, se non avete troppa fretta, il consiglio è quello di attendere per vedere quali saranno le proposte della Commodore. Se invece non potete attendere, le soluzioni oggi sono le seguenti: 1) Per i possessori dell'A500 - acquistare l'A570; 2) Per tutti la linea Amiga - acquistare un CDTV, interfacciarlo all'Amiga tramite la porta parallela e il programma di pubblico dominio ParNet; 3) Per tutta la linea Amiga - Acquistare un controller SCSI, un drive CD-ROM SCSI compatibile e un driver di gestione (per il discorso sul driver vi rimandiamo a pagina 59-60 di questo stesso numero).

**Indirizzate
tutta la corrispondenza
per la rivista a:**

COMMODORE GAZETTE
La posta della Gazette
Via Monte Napoleone, 9
20121 Milano

Preghiamo i lettori di essere concisi e concreti, per darci modo di rispondere al più grande numero possibile di lettere. La redazione si riserva comunque il diritto di sintetizzare le lettere troppo lunghe.

Amiga 1200 è molto più potente e veloce del vostro Amiga 500.



Perché non fate cambio?

**Se acquistate Amiga 1200, vi supervalutiamo
il vostro Amiga 500.**

Veloci, l'offerta è valida fino al 30 giugno.

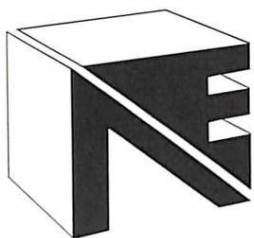
Ragazzi, non c'è tempo da perdere. Prendete il vostro Amiga 500 e portatelo in uno dei Commodore Point che partecipano all'iniziativa (se non sapete dov'è, telefonate al numero verde). Su ogni Amiga 500 funzionante, c'è una supervalutazione di 200.000 lire. E in più,

**NUMEROVERDE
1678-27012**

vi diamo un computer ancora più super: Amiga 1200*. Il più potente,
il più veloce, nessuno lo batte. E allora, perché non fate cambio?

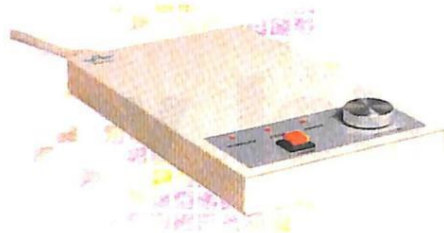
Commodore
FACILE IL DIFFICILE.

*lire 790.000 (IVA inclusa)

**NEWEL® srl****COMPUTERS ACCESSORI VIDEOGAMES**
20155 MILANO - VIA MAC MAHON 75**TEL. NEGOZIO (02) 39260744 (5 linee r.a.) FAX 24 ORE (02) 33000035 (2 linee r.a.)****ORDINA SUBITO**
02 - 33000036 (5 linee)**VENDITA
ANCHE PER
CORRISPONDENZA
IN TUTTA
ITALIA**

ROCKGEN L. 279.000

Genlock amatoriale con controllo di dissolvenza per la sovrapposizione dell'immagine di Amiga. Passante video automatico. Alimentazione da computer o da fonte esterna. Compatibile con tutti gli Amiga compreso il Commodore CDTV, oltre ad avere una totale compatibilità con i sistemi Pal/NTSC.



ROCKGEN PLUS

L. 449.000

Genlock semiprofessionale con regolazioni di fader, mode e invert. Indicatore di segnale Video presente. Alimentazione ad Amiga e/o esterna. Dissolvenza duale con due manopole per la regolazione dell'overlay e invert effect. RGB indipenden-

te e passante video pass-thru per separare il segnale Amiga da quello video. Ingresso key-in per dispositivi croma. Compatibile con tutti gli Amiga e Commodore VDTV e compatibilità dei sistemi video Pal/NTSC.



SUPER MAXIGEN L. 990.000

Nuovissimo genlock professionale, qualità Broadcast con S_VHS in uscita, regolazione livelli, 2 uscite video per visualizzare il vostro lavoro mentre viene registrato. Possibilità di Super impose. Banda passante 6 MHz. 1 Vpp 75 Ohm. Serie di effetti video e manuale in italiano. Alimentazione esterna a 500mA 12V (alimentatore fornito).

ROCKEY L. 699.000

Questo utile accessorio è simile ad un genlock ma il suo compito è fare proprio l'opposto. A differenza dei normali Genlock, che sovrappongono l'immagine di Amiga su una fonte video, questo estrae ad esempio un gatto che cammina in mezzo ad una stanza e lo sovrappone, eliminando lo sfondo della stanza, ad una qualsiasi pagina grafica di Amiga. Effetti speciali per Sandwich e inverso per produzioni Amiga, RGB splitter incorporato per applicazioni di digitalizzazioni. Lavora in abbinamento con la maggior parte dei comuni Genlock. Compatibile con la porta video RGB di Amiga a 15 Hz oltre ad avere un passante RGB, passante video e porta Key-In. Compatibile con i formati video Pal/NTSC.

DVE 10-P L. 1.990.000

Sofisticato Genlock per tutti gli Amiga. Ingressi S-VHS e videocomposito. Mixer video e audio. Effetti tenda speciali per la visualizzazione delle varie fonti d'ingresso, Pin-P, effetti Video digitali, Video processore, digitalizzatore video. Il DVE-10P e Amiga trasforma il vostro sistema in un laboratorio per titolazioni, animazioni e grafici.

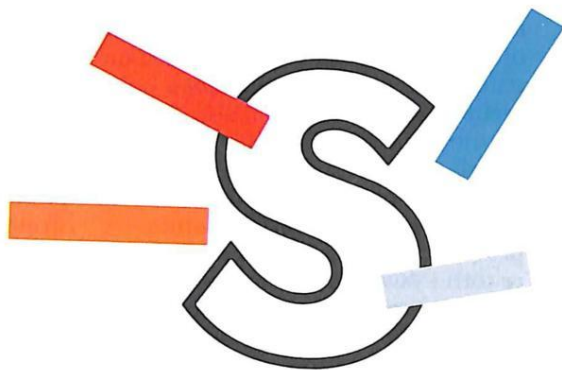
P.I.P. VIEW L. 299.000

Il P.I.P. VIEW è un dispositivo elettronico esterno con telecomando collegabile ad un monitor Amiga con presa Video-In (Commodore 1084-Philips 8833 ecc.), il quale permette di ricevere e visualizzare i programmi televisivi sul Vostro monitor. Inoltre è dotato del sistema pinP (Picture in Picture) che permette di vedere contemporaneamente il Vostro film preferito e la partita di calcio della Vostra squadra. È dotato di tre ingressi video e uscita video e 1 uscita audio.



SOFTWARE GALLERY

UNA GUIDA PER ORIENTARSI NEL MONDO DEL SOFTWARE



ART NOUVEAU

Un nuovo programma italiano di grafica pittorica

Computer: Amiga
Supporto: Disco
Prezzo: L. 79.900
Produzione/distribuzione: Digiteam
(Via G. Modena 9, 20129 Milano -
☎ 02/29409073)

GIUDIZIO
COMPLESSIVO:
MEDIOCRE



Funzionalità:	★	★					
Conferma aspettative:	★	★					
Documentazione:	★	★	★	★	★		
Prezzo/prestazioni:	★	★	★				

In questo periodo stiamo assistendo a una proliferazione di prodotti italiani nel campo della grafica pittorica: sono infatti appena usciti l'ottimo *Personal Paint* della Cloanto e la nuova versione di *Progetto Immagine* della Menti Possibili. Anche la Digiteam s'inserisce in questo filone con un nuovo prodotto: *Art Nouveau*.

Il programma viene commercializzato in una confezione contenente un dischetto, la cartolina di registrazione e un manuale in italiano di circa sessanta pagine. Il manuale è ottimo:

SCHEDA CRITICA



INSUFFICIENTE (★)
Un pessimo prodotto che non merita nessuna considerazione.



MEDIOCRE (★★)
Il programma ha alcuni difetti di fondo, anche se nel complesso raggiunge quasi la sufficienza.



SUFFICIENTE (★★★)
Un prodotto accettabile, ma non aspettatevi grandissime emozioni.



DISCRETO (★★★★)
Un programma desiderabile, ma c'è sicuramente di meglio.



BUONO (★★★★★)
Raccomandato vivamente: tra i migliori programmi della sua categoria.



OTTIMO (★★★★★★)
Eccezionale! Fino a oggi non si era mai visto nulla del genere.

scritto in buon italiano, molto chiaro, ben suddiviso in brevi capitoli. Si nota un'attenzione nei confronti dell'utilizzatore meno esperto, ma non mancano informazioni dettagliate che si rivelano utili anche per gli utenti più smaliziati. Il programma dovrebbe funzionare su qualsiasi macchina Amiga, NTSC o PAL, con qualsiasi quantità di memoria e supporta i nuovi chip AGA. Grazie a speciali librerie, l'interfaccia del programma si mantiene praticamente uguale con qualsiasi versione di *Kickstart*, dalla 1.2 in poi, ed è interamente in italiano. *Art Nouveau* può facilmente essere installato su hard disk semplicemente copiando dal dischetto il cassetto che lo contiene.

Alla partenza del programma, viene visualizzato uno "screen requester" tipico del sistema 2.0 (se la versione usata del sistema è precedente alla 2.0, questo requester viene comunque simulato), mediante il quale si può selezionare il tipo di schermo che il programma deve aprire. Una volta scelto il tipo di schermo, si entra nel vero e proprio ambiente di lavoro. Qui si evidenziano subito alcune differenze d'impostazione rispetto agli altri programmi pittorici: per esempio, tutte le icone di selezione degli strumenti di disegno sono contenute in vere e proprie finestre *Intuition*, con tanto di barra di riposizionamento e di gadget di profondità. Il programma permette di attivare i seguenti tipi di finestre: attrezzi, colori, ritagli, retini, zoom e coordinate.

La finestra degli attrezzi contiene la maggior parte delle icone di disegno ormai usuali in qualsiasi programma pittorico: disegno di linee continue e punteggiate, di rettangoli pieni e vuoti, di ellissi piene e vuote, disegno con l'aerografo, riempimento di aree chiuse, disegno di spezzate, scrittura di testi, cancellazione dell'ultima modifica o di tutto il disegno. Un po' meno usuali sono le opzioni di ribaltamento del disegno in verticale od orizzontale, di cattura dello schermo sottostante al programma, di spostamento mediante il mouse dell'immagine nel caso che lo schermo non fosse sufficiente a contenerla per intero. Interessante è poi la possibilità di disegnare curve di Bezier che permettono di creare archi molto

dei propri programmi. Tra l'altro, il codice C generato è compatibile X-Windows. Ma non è finita qui: il ritaglio può essere salvato anche come icona e se ne possono definire alcune caratteristiche, come il tipo oppure il tool associato. Il programma può quindi essere usato in questo caso come un potente icon editor. *Art Nouveau* permette poi di ricaricare da disco i ritagli in tutti i formati prima elencati: è quindi possibile caricare un'immagine a partire da un sorgente C o Assembly!

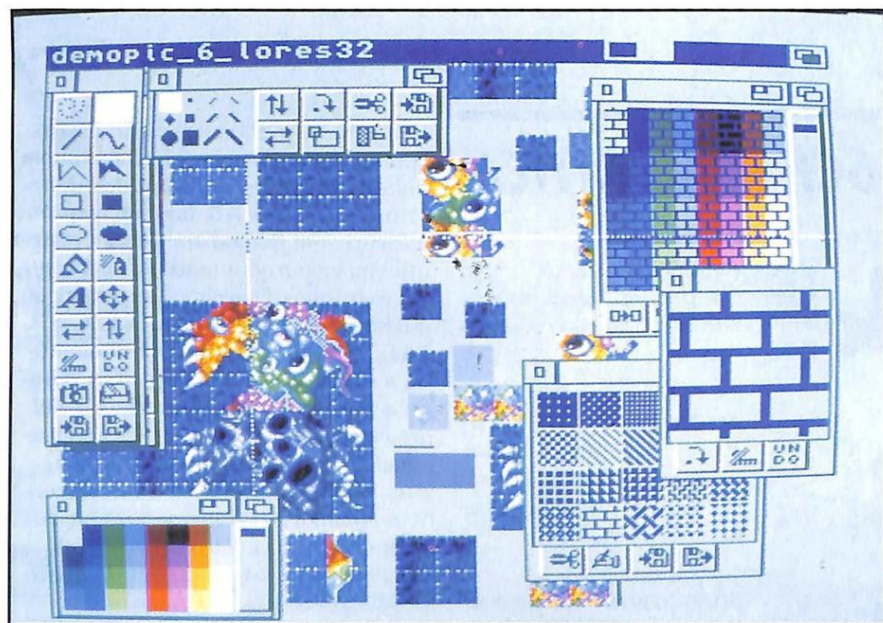
Purtroppo, non abbiamo potuto provare questa capacità di poter gestire tutti i vari formati, che ci sembra essere il punto di forza del programma, dato che il demo messoci a disposizione per il test non permetteva di

sistema 3.0, per modificare i colori verrà utilizzata la "ruota dei colori" tipica di questa versione di sistema. Nella finestra dei colori compare però anche uno slider che permette di miscelare in tutti i modi possibili i colori con il retino precedentemente scelto: si può passare quindi gradualmente da colori "puri" a colori completamente "retinati". In questo modo, i retini possono anche essere usati per creare colori apparenti non originalmente presenti nella palette dello schermo, sfruttando la giustapposizione di colori vicini. In effetti, se si prende un po' la mano con questo strumento, si possono raggiungere risultati interessanti.

La finestra di zoom permette d'ingrandire una porzione rettangolare dell'immagine fino a 12 volte e di editarla pixel per pixel; non è possibile utilizzare in questa finestra gli altri attrezzi di disegno.

Infine, la finestra delle coordinate permette di visualizzare le coordinate correnti del cursore sia assolute sia relative a un'origine definita dall'utente.

Molte delle opzioni presenti nelle finestre sono duplicate nei menu; da menu è inoltre possibile aprire o chiudere le varie finestre, attivare una griglia di riferimento scalabile che permette di eseguire disegni regolari, ottenere informazioni sulla memoria disponibile, attivare o meno una seconda pagina di schermo dove poter effettuare eventuali prove di disegno, selezionare una fonte per la scrittura di testi o iconizzare l'intero programma per poterlo poi riattivare quando lo si desidera. Inoltre, a ciascuna opzione corrisponde un tasto della tastiera: l'elenco dei tasti utilizzabili è riportato nell'appendice del manuale. Vi sono poi alcuni tasti che permettono di effettuare operazioni non presenti nei menu o nelle finestre. Il tasto "spazio", per esempio, seleziona una palette di colori che permette di vedere bene le icone delle finestre; se si preme lo "spazio" una seconda volta si ritorna alla palette originale dell'immagine (preferiamo comunque la soluzione adottata da *Personal Paint*, che cerca automaticamente i colori della palette dell'immagine più adatti per la visualizzazione delle icone). Un altro tasto utile è "Del", con il quale si possono far scomparire dallo schermo tutte le finestre in modo da poter lavorare sull'immagine intera; premendolo di nuovo ricompariranno le finestre atti-



precisi. Sempre da questa finestra si può inoltre salvare o caricare un'immagine in formato ILBM.

La finestra dei ritagli permette appunto di ritagliare un'area rettangolare dello schermo per poi modificarla in vari modi, per esempio ruotandola di 90 gradi oppure cambiandone le dimensioni. Tale area potrà poi essere "stampata" sullo schermo premendo il tasto del mouse. Una caratteristica importante del programma è che questo ritaglio potrà essere salvato su disco in vari formati: oltre al solito ILBM, può essere salvato in formato BitMap (cioè esattamente come viene rappresentato in memoria), oppure salvato sotto forma di codice C o di codice Assembly, in modo da poter immediatamente utilizzare una certa immagine all'interno

salvare alcun file, né ci sono stati forniti esempi di sorgenti C creati da *Art Nouveau*. Su quest'aspetto non possiamo quindi che riportare le informazioni tratte dal manuale.

Un'altra finestra interessante è quella dei retini, con la quale si possono selezionare vari tipi di motivi grafici che possono venire utilizzati per disegnare. Questi retini possono anche essere salvati e ricaricati da disco, editati punto per punto oppure estratti da un ritaglio.

Una volta selezionato un retino, si deve far uso della finestra dei colori. In essa è possibile modificare tutti i colori della palette nei modi usuali (cursori R G B, scambio, di colori, sfumature tra due colori, eccetera): è da notare che in caso si disponga del

ve precedentemente.

Tutto questo viene fatto in soli 87K di programma: la Digiteam ha infatti scritto molte parti del codice direttamente in Assembly, riuscendo quindi a contenere la dimensione del programma. Questo particolare può risultare utile sia per chi non dispone di un hard disk e vuole un programma pittoresco che non occupi un dischetto intero, sia per chi non dispone di molta memoria e vuole per esempio lasciare sempre attivo il programma per catturare alcune schermate.

I lati negativi e le conclusioni

Finora abbiamo parlato degli aspetti positivi di *Art Nouveau*, ora dobbiamo purtroppo elencare quelli negativi. Innanzitutto, le finestre delle opzioni non vengono scalate opportunamente a seconda della risoluzione dello schermo. Il risultato pratico è che quando si è in bassa risoluzione le finestre risultano alquanto ingombranti e se si devono utilizzare molte opzioni si finisce per coprire quasi completamente il disegno; se si è in alta risoluzione interlacciata le icone delle finestre diventano talmente piccole da essere praticamente indistinguibili le une dalle altre. Mentre nel primo caso si può porre rimedio al problema facendo uso ripetutamente del tasto "Del" prima descritto, nel secondo caso non sembra esserci soluzione, a meno di non armarsi di lente d'ingrandimento!

Anche il fatto che le opzioni siano sparse su più finestre invece di essere raggruppate tutte su una sola sconcerta l'utilizzatore, soprattutto all'inizio. In pratica, per utilizzare il programma agevolmente bisogna memorizzare i tasti corrispondenti alle opzioni più usate.

Nel manuale si parla di un file di definizione delle opzioni di default del programma; in esso si dovrebbe poter specificare per esempio quali finestre attivare all'inizio o quali retini caricare da disco. Sembra però che, almeno nella versione messaci a disposizione per il test, questo file venga completamente ignorato dal programma, che quindi parte in ogni caso con dei default interni.

Un'altra cosa che non siamo assolutamente riusciti a fare è caricare immagini in formato HAM, nonostante il manuale dica che *Art Nouveau* dovrebbe permettere almeno di visualizzarle,

ma non di modificarle. Colpisce poi la lentezza di alcune operazioni, come la scalatura di un ritaglio o il tracciamento di linee con un pennello che non sia il singolo punto.

Ci è anche capitato che, dopo aver effettuato un po' di operazioni, all'uscita il programma andasse in crash, denunciando qualche piccolo bug probabilmente in fase di rilascio delle risorse allocate.

Ma quello che più mette in difficoltà questo programma è il confronto con i suoi concorrenti, in particolare *Personal Paint*, recensito sul numero 1/93 di *Commodore Gazette*. Con una differenza di prezzo di sole 20 mila lire *Personal Paint* offre moltissime opzioni in più. Solo per citarne alcune, in *Art Nouveau* è assolutamente assente un'elaborazione digitale dell'immagine di qualsiasi tipo, mancano molte delle opzioni di elaborazione del colore offerte da *P-Paint*, non è possibile scalare automaticamente un'immagine cambiando il formato della pagina, non è possibile utilizzare i tool di disegno nell'area di zoom, eccetera. Senza contare che, almeno a nostro giudizio, l'interfaccia di *P-Paint* risulta di più immediato utilizzo rispetto a quella di *Art Nouveau*. Ovviamente, anche il confronto con un programma come *Deluxe Paint IV* non regge, ma in questo caso la differenza di prezzo è sostanziale.

A vantaggio di *Art Nouveau* rimangono solo quelle peculiarità che abbiamo già evidenziato in precedenza, come i formati di salvataggio e di caricamento dei ritagli, l'uso dei retini, le curve di Bezier e l'estrema compattezza del programma, che lo rendono ideale più per il programmatore che deve manipolare delle immagini nei propri programmi che per l'artista che si propone di creare dei capolavori.

Forse, almeno per questa prima versione del programma, si poteva pensare a un costo ancora più ridotto per invogliare all'acquisto quei programmatori che attualmente fanno uso di varie utility, magari meno potenti, per poter inserire nei propri programmi immagini o per crearsi le proprie icone.

Ci auguriamo comunque che la Digiteam preveda entro breve tempo una nuova versione di *Art Nouveau* per rimediare a quei difetti, come la mancata scalatura delle finestre di opzioni, che attualmente sminuiscono il valore di questo programma.

D.D.

B-17 FLYING FORTRESS

Bomba o non bomba...

Computer: Amiga
Supporto: Disco
Prezzo: L. 99.900
Produzione: Microprose
Distribuzione: Leader (Via Adua 22, 21045 Gazzada Schianno - ☎ 0332/874111)

GIUDIZIO
COMPLESSIVO:
BUONO



Grafica:	★★★★★
Sonoro:	★★★★
Giocabilità:	★★★★★
Prezzo:	★★

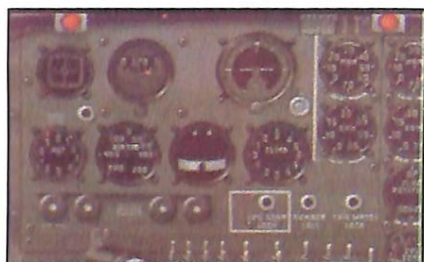
La Microprose è padrona del cielo ormai da tempo: di qualsiasi periodo (bellico) storico si parli, possiamo stare sicuri che la casa americana ha la sua simulazione di volo relativa. Nella storia del software targato Microprose, sono stati coperti tutti i possibili conflitti che coinvolgessero forze aeree: si va dai triplani della Prima guerra mondiale agli F-117 Stealth moderni.

Ma il progetto stavolta è ambizioso e particolare: porvi alla guida di un solo aereo con l'impegno d'impersonare e guidare i dieci elementi dell'equipaggio, ognuno con le sue caratteristiche e il proprio ruolo all'interno del velivolo. Il gioco in questo senso ha anche un pizzico di caratteristica strappata ai giochi di ruolo: l'attitudine complessiva di ogni singolo membro dell'equipaggio viene scorporata in cinque parametri base, che sono Mitragliamento, Bombardamento, Conoscenze tecniche, Conoscenze mediche e Pilotaggio. Alla fine di ogni missione riuscita, questi parametri si alzeranno facendo così "evolvere" il personaggio.

L'equipaggio di un B-17 è composto di nientemeno che dieci persone: abbiamo un bombardiere, un navigatore, un pilota, un copilota, un radiofonista e cinque mitraglieri. Ovviamente, pilota e copilota si occupano della guida dell'aereo e hanno sott'occhio la strumentazione.

Essendo *B-17 Flying Fortress* particolarmente realistico, ci si dovrà occupare della potenza dei quattro motori singolarmente (tasti 1-4)! C'è persino l'estintore dei motori da attivare in caso d'incendio. Il bombardiere ovviamente si occupa del rilascio delle bombe e deve avere una particolare mira. Il navigatore dev'essere bravo a mantenere la rotta stabilita verso e via dal bersaglio. Il radiofonista ha una serie di messaggi in un apposito menu che può inviare: aereo a ore 12, membro dell'equipaggio ferito e così via. I mitraglieri occupano le postazioni dotate di mitragliatrici del bombardiere (in totale sei).

Ma forse qualcuno si starà chiedendo se tutto questo non sia eccessiva fedeltà al realismo a discapito dell'agilità di gestione dell'equipaggio del *B-17*. No, direi piuttosto che è un'occasione per sperimentare una diversa attività e che tutto è estremamente divertente e dà un senso dell'attività di



squadra davvero unico per un gioco di guerra. Il realismo, poi, è al solito a grandi livelli: nel gioco sono stati compresi i modelli di caccia vi sareste trovati addosso se aveste volato sull'Europa in quel periodo. La grafica ha la sua minima quantità di scatti nel caso di paesaggi troppo affollati di poligoni, ma giocando un po' con il livello di dettaglio, si giunge a compromessi accettabili.

Nel complesso, *B-17 Flying Fortress* rappresenta qualcosa di veramente unico nel panorama delle simulazioni di volo: è il primo simulatore in cui sperimenterete la vita di un intero equipaggio senza per questo doversi trovare in affanno per passare da un ruolo all'altro.

Dopo un po' di partite, tutto riesce con estrema naturalezza e vi accorgete che *B-17* genera un "mitico" spirito di squadra, qualcosa di davvero mai visto in un simulatore. Un programma che mi sento quindi di consigliare senza riserve.

P.C.

WORLD NEWS

Novità sull'Amiga da tutto il mondo

a cura di Marco Dufour

Se ci si ferma un attimo a pensare alla costante evoluzione della linea Amiga, alla mole di programmi e soluzioni hardware dedicati a questo computer, è impossibile non rimanere impressionati. Più volte si è sentito parlare della storia dell'Amiga, nato col nome Lorraine nei laboratori della HiToro, e spesso negli editoriali delle riviste oltreoceano si è letta una certa apprensione per il futuro dell'Amiga. La storia fortunatamente ci sta dando ragione e a chi vedeva questa linea della Commodore ormai defunta da tempo, basta presentare la lista delle novità che ogni mese caratterizzano il mercato. Se non fossero stati introdotti i nuovi modelli (A4000 e A1200), la situazione forse sarebbe diversa: a meno di un ribassamento dei prezzi molto sostenuto, i vecchi modelli, proprio perché "vecchi", non avrebbero retto la concorrenza ancora a lungo. Era ovvio che il mercato si aspettava dei nuovi modelli, come del resto accade tutt'ora, un po' meno ovvio che la Commodore sarebbe riuscita a produrli in tempo e con caratteristiche tanto interessanti. Prendiamo in considerazione l'Amiga 4000: processore 68040 a 25 MHz, 6 MB di RAM, drive HD, gestione dei colori a 24 bit, bus di espansione a 32 bit a 25 MHz, sistema operativo multilingua e multitasking. La lista potrebbe continuare a lungo, perché le doti di questo modello non sono poche, ma ora andiamo a vedere cos'altro offre il mercato. Se rimaniamo in ambiente Motorola, il paragone che subito viene da fare è con i modelli Macintosh della Apple. Le caratteristiche dell'Amiga 4000 si possono solamente trovare sui modelli Quadra che vengono a costare dagli 8 milioni in su. Cosa dire poi dei PC, dove per avere il bus di espansione a 32 bit a 25 MHz, il prezzo viene maggiorato di quasi un milione? Sono forse considerazioni di poco conto, ma che fanno pensare che dal punto di vista tecnico alla Commodore fanno sul serio. Non voglio assolutamente dire che l'Amiga sia la macchina perfetta, ma che quanto a caratteristiche tecniche, prezzo ed espandibilità è difficile trovare qualcosa di più attraente sul mercato. Modelli nuovi sono già usciti, altri presto usciranno (CDTV2, nuovo A3000 con chip AGA), smentendo appieno chi pensava che il discorso Amiga fosse ormai chiuso.

Un grande dilemma assale ora quasi tutti gli utenti Amiga: fare o non fare l'upgrade alle nuove macchine? Quasi tutti gli utenti almeno una volta si sono posti questo interrogativo, cercando di trovare una possibile soluzione. A rendere ancora più difficile questa scelta è la continua introduzione di soluzioni hardware in grado di portare i vecchi modelli alle prestazioni dei nuovi. Avevamo già visto il mese scorso la scheda DAC18 (commercializzata dalla Newtronic in collaborazione con la VideoPress), in grado di ottenere 260 mila colori su qualsiasi modello Amiga. Anche per quanto riguarda le schede acceleratrici molte cose si sono messe a produrre di nuove. La soluzione studiata dalla Supra (7101 Supra Drive SW, P.O. Box 7101, Albany, OR 97321-8000, USA, Tel. 001/503/9672410) ha dell'incredibile: mentre tutte le società si sono lanciate sui nuovi processori a 32 bit della Motorola, la Supra ha optato per una versione super veloce del 68HC000. Questo vuol dire che avremo a disposizione un processore 16 bit a 28 MHz. Disponibile per Amiga 500 e Amiga 2000, SupraTurbo 28 (199 dollari, circa 300 mila lire) è in grado di velocizzare le operazioni del sistema operativo di circa 4 o 5 volte, permettendo per esempio il multitasking di programmi di telecomunica-

zione molto veloci con altro software, e la decompressione veloce di animazioni e immagini. Quello che risulta comunque maggiormente interessante, è il fatto che essendo basata su un processore della famiglia 68000, offre la più totale compatibilità con tutto il software progettato per funzionare sulle vecchie macchine; la scheda è comunque disattivabile tramite un interruttore esterno o un programma appositamente studiato.

Dalla GVP, invece, arrivano notizie riguardanti le nuove schede per Amiga 1200, la **A1230 Turbo+** e la **A1200 SCSI/RAM+**. La A1230 Turbo+ è una scheda da inserire nello slot interno dell'A1200 (che permette di non invalidare la garanzia), che offre un processore 68EC030 a 40 MHz, 1 MB di RAM a 60 nanosecondi espandibile a 32 MB e la possibilità di aggiungere un coprocessore matematico. Il prezzo di questa scheda è stato fissato a 399 dollari (circa 600 mila lire), comprensivo di software per rimappare il Kickstart in RAM. La scheda A1200 SCSI/RAM+, per la quale non è ancora stato fissato un prezzo definitivo, offre un controller per hard disk SCSI (circa tre volte più veloce di quello presente sull'Amiga 1200), 1 MB di RAM a 32 bit (60 nanosecondi) espandibile a 8 MB e uno zoccolo per il coprocessore matematico (opzionale).

Un interessante prodotto viene proposto dalla ICD: il suo nome, **Trifecta**, può già suggerire la multifunzionalità di questa scheda. Si tratta infatti di un controller in standard AT-IDE e SCSI 2 con la possibilità di montare espansioni di memoria (accetta fino a 8 MB di RAM attraverso ZIP DRAM). Stando a quanto ci riferisce la ICD (1220 Rock St., Rockford, IL 61101, USA, Tel. 001/815/9682228), il controller SCSI 2 montato offre una velocità di trasferimento pari a 10 MB al secondo in modo sincrono o 5 MB in modo asincrono. Il controller lavora in pieno supporto DMA, lasciando libero il processore centrale dalle funzioni di lettura dell'hard disk. Come già detto, Trifecta offre anche un controller AT-IDE: se l'utilizzo di un hard disk risulta indispensabile, ma le disponibilità economiche non permettono l'acquisto di un disco SCSI (più veloce, ma anche più costoso), è possibile collegare, anche provvisoriamente, un disco in standard IDE. Si avrà quindi la possibilità di sfruttare tutte le vantaggiose caratteristiche di un hard disk mantenendo un prezzo contenuto, senza rinunciare alla possibilità di collegare unità SCSI esterne o interne. Trifecta 2000IX costa 250 dollari (circa 350 mila lire), mentre la versione per Amiga 500 (denominata Trifecta 500XI) costa 350 dollari (circa 525 mila lire). Avevo accennato nel numero scorso all'esistenza di diversi driver per stampante in grado di soddisfare le esigenze di quasi tutti gli utenti. A questo proposito segnalo la presenza su alcune BBS di nuovi driver per la stampante HP 550C. La **Inovatronics** (Im Heidkamp 11, W-5000 Cologne 91, Germany, Tel. 0049/221/875126) ha immesso sul mercato **TurboPrint 2.0** un programma in grado di pilotare quasi tutte le stampanti esistenti sul mercato. La caratteristica di questo programma è la sua grande configurabilità, sia per la stampa di testo, sia per le immagini bitmap. Per migliorare la qualità di stampa, sfrutta un sistema denominato "mezza linea" in grado di ridurre le bande orizzontali e due nuovi algoritmi di smoothing per lo smussamento dei pixel. Sono disponibili 14 tipi di dithering diversi per la stampa delle immagini in tonalità di grigio o a colori: per questi è possibile anche decidere la rotazione e le dimensioni. Si possono stampare cartelloni di enormi dimensioni grazie all'opzione "Tile" e stampare in negativo, regolando anche luminosità e contrasto. **TurboPrint 2.0** costa 129 dollari (circa 200 mila lire).

La **Epson** finalmente riconosce le potenzialità dell'Amiga e oltre a pubblicizzare i suoi prodotti su alcune riviste dedicate ai modelli Commodore (negli USA), ha deciso il pieno supporto per i nuovi scanner a colori 600C e 800C. Entrambi i modelli vengono collegati tramite porta SCSI e pilotati dal driver di *Art Department Professional 2*. Offrendo

SUPERFROG

Il miglior platform della storia!

scansioni in singola passata a 16 milioni di colori ad alte risoluzioni (fino a 600 dpi, punti per pollice), questi scanner risultano la soluzione ideale per chi deve fare fotoritocco e impaginazione elettronica. Sempre che il vostro interesse sia rivolto al desktop publishing, vi segnaliamo un disco pieno di **Genies** per *Professional Page 3* (e superiori) commercializzato dalla DevWare (12520 Kirkham Court, Suite 1-AW36, Poway, CA 92064, USA, Tel. 001/619/679/2825). Il disco in questione fa parte di un'enorme collezione di programmi di pubblico dominio ed è denominato **WB121: ProPage3 Enhancer**; nel disco sono contenuti oltre 40 script **ARexx** (i "Genies") in grado di automatizzare l'utilizzo di *Professional Page*. Che dire per esempio della macro per ridimensionare il testo per farlo stare perfettamente all'interno di un box? O la possibilità di generare grafici a torta direttamente dal programma? Il costo è molto ridotto (6 dollari, circa diecimila lire) e se il contenuto è realmente così interessante, è un vero peccato non procurarselo.

In Germania la **Boeder**, una catena di distribuzione di prodotti per computer, ha immesso sul mercato la nuova linea **VECTOR**. Nel catalogo sono presenti prodotti come espansioni di memoria, ROM switcher, cavi seriali e paralleli, modem, mouse, tutto dedicato alla linea Amiga. Oltre alle soluzioni hardware (vengono vendute anche espansioni PCMCIA e SRAM, memoria non volatile), vengono distribuiti diversi programmi. Tra questi sembra molto interessante **Translator2**, programma in grado di tradurre frasi dal tedesco all'inglese e viceversa; è prevista anche una versione comprendente il francese e forse anche l'italiano. La linea di prodotti Boeder dovrebbe essere disponibile nei grandi magazzini METRO.

La **Micronik Computer Service** (Tel. 0049/214/93186 - fax 95791) offre un case **Bi-gTower** per Amiga 3000 e 4000, in grado di ospitare fino a 6 unità dischi. Sarà presto disponibile anche il modello **MiniTower** per Amiga 600 e 1200. Questi modelli, esteticamente ben riusciti, offrono la possibilità di espandere il proprio Amiga ai limiti delle capacità della macchina. Il prezzo non è ancora stato fissato, ma dovrebbe aggirarsi attorno ai 450 marchi (versione per A3000/4000) e 550 marchi nella versione per Amiga 600 e 1200 (circa mezzo milione di lire).

Gli appassionati più attenti si saranno accorti che sulle nuove scatole degli Amiga 4000 è presente la dicitura "68030". Questa versione, caratterizzata da un prezzo ridotto rispetto alla versione con processore Motorola 68040 è appunto denominata A4000/030. Attualmente, l'Amiga 4000/030 è già disponibile anche in Italia e offre 4 MB di RAM (due di memoria chip, due di memoria fast), un hard disk da 80 MB (in standard AT-IDE) e un processore 68EC030 (senza MMU), al prezzo di 2.900.000 lire iva compresa. Non si è ancora capito se questa versione sia il tanto vociferato Amiga 2400 o semplicemente un modello di transizione per provare la nuova scheda '030. Risulta invece significativa la possibilità di espandere in fasi successive questo computer, fino a raggiungere tutte le caratteristiche di un vero Amiga 4000. Ci si aspetta presto anche il modello Tower dell'Amiga 4000 e il nuovo controller standard SCSI2, denominato A4091.

Forse entro l'estate sarà data la notizia ufficiale del supporto del **Photo CD** della Kodak per il CDTV. Le difficoltà riscontrate fino a questo punto per sfruttare questa caratteristica su CDTV sono dipese da problemi di copyright più che da problemi tecnici. Torna quindi la voce che la Commodore abbia finalmente preso accordi con la Kodak e che prossimamente potremo utilizzare il computer Commodore anche per vedere le foto nel formato Kodak, con la possibilità di operare effetti di fotoritocco. Il prossimo venturo **CDTV2**, basato sulla scheda madre dell'Amiga 1200, disporrà anche delle risoluzioni grafiche ideali per visualizzare fotografie in true colour.

Computer: Amiga
Supporto: Disco
Prezzo: L. 79.900
Produzione: Team 17
Distribuzione: Leader (Via Adua 22, 21045 Gazzada Schianno - ☎ 0332/874111)

GIUDIZIO
COMPLESSIVO:
OTTIMO



Grafica:	★★★★★★
Sonoro:	★★★★★★
Giocabilità:	★★★★★★
Prezzo:	★★

Avevamo appena finito di tessere le lodi di *Body Blows*, il miglior picchiaduro mai apparso sull'Amiga, che subito il miracolo avviene anche per la categoria piattaforma: davvero non ci sono più parole per definire la capacità e il talento dei Team 17, ormai diventati le vere star dell'universo videoludico dell'Amiga. Anche *Superfrog*, infatti, entra di diritto nell'olimpo dei più grandi giochi mai realizzati per il 16 bit della Commodore.

Ma andiamo con ordine: *Superfrog* è un platform la cui protagonista è una super-rana (capirete la genesi del personaggio dopo esservi gustati la demenzialissima introduzione) le cui primarie capacità sono quelle di spiccare salti notevoli e di procedere con un passo speditissimo. Questo naturalmente gli servirà per accedere alle varie altezze a cui si trovano le piattaforme del gioco.

Il gioco propone cinque livelli, più uno segreto con lo scontro finale contro l'arcinemico (la strega che vi ha trasformato in rana), che rappresenta praticamente il meglio che ci si potrebbe aspettare da un prodotto di questo genere. Ogni livello è infatti letteralmente ricolmo di bonus e sotto-bonus che donano punti, ma anche nuove abilità alla ranocchia. Raccogliendo un piccolo esserino tondo, sarete poi in grado di lanciarlo verso i nemici mettendoli così definitivamente k.o. Incamerando le ali, avrete la possibilità di fluttuare per aria (tramite

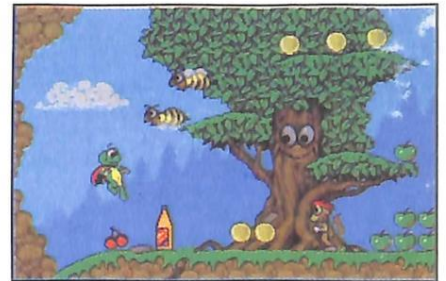
pressione ripetuta del pulsante) dopo un salto; appropriandovi di una pillola rossa avrete il dono dell'invisibilità, che corrisponderebbe poi all'invulnerabilità: solo gli uschioni del batrace rimarranno in vista!

Altro elemento fondamentale per un platform moderno devono essere oggetti, passaggi e stanze segrete. Ebbene, *Superfrog* si dimostra perfettamente attrezzato, con una serie di cunicoli, canali di collegamento e locazioni segrete che aspettano solo di essere portati alla luce.

L'ambientazione è classicamente favolistica. A parte la simpatica rana, abbiamo nemici tipicamente fumettistici, tutti contraddistinti dai due occhioni che suggeriscono praticamente il "teneroso" (come si usa dire in gergo) stile grafico del gioco.

Il look generale è davvero splendido e ognuno dei livelli ha la sua bella quantità di dettagli fumettistici; quel che colpisce è lo scorrimento che riesce a scattare fluidamente anche alla massima velocità della rana.

Che dire poi della musica? Un mix costellato di gracidi davvero da ascoltare. Se questo non bastasse, ci sono anche la slot machine che fa guadagnare vite, punti e addirittura il codice



del livello corrente, e uno sparattutto nascosto che fa capolino verso la fine del gioco.

In definitiva, per *Superfrog* non si può che gridare al capolavoro: tutto il meglio che si potesse pretendere da un videogioco di piattaforma è concentrato in questo titolo.

Né grafica, né sonoro, né giocabilità potrebbero passare inosservati quando si tratta dei Team 17, e ne vedremo ancora delle belle. I prossimi appuntamenti sono infatti con titoli come *Overhead Racing* e *Alien Breed 2* (solo sull'Amiga 1200, però). Arrivererci quindi al prossimo prodotto di questo ormai mitico "team", che non possiamo che lodare.

P.C.

PRODUCTIVITY UPDATE

Ogni mese vengono pubblicati decine di nuovi programmi e aggiornamenti di versione. Non tutte le versioni possono essere provate sulla rivista e comunque non in tempi brevi. In ogni numero vi forniremo un quadro il più possibile esauriente e aggiornato sulle ultime novità e nuove versioni immesse sul mercato. Le varie versioni sono da considerarsi finali e disponibili al pubblico, pertanto i comunicati stampa delle software house, le anticipazioni, le pre-release o beta test, non sono considerati. I nuovi programmi e gli aggiornamenti sono indicati in nero maiuscolo. La denominazione AGA indica che il programma supporta i modi grafici introdotti col nuovo chipset.

PROGRAMMA RELEASE VERS.

3D Professional	1.13	PAL
Adorage	1.81d	
Advantage	1.1	
Aegis Sonix	1.3	
Aegis Visionary	1.0	
ALADDIN 4D	2.1	
A-Max II	2.53b	
Ami-Back	2.0e	
AmigaTex	3.1a	
Amiga Vision	1.70 Rev. z	
Amiga Vision Professional	1.0 (AGA)	
Amos	1.36	
Amos Compiler	1.2	
Amos Professional	2.0	
AMPLIT	2.0	
Animaker	1.1	
Animation: Apprentice	1.0	
Animation: Editor	1.0	
Animation: Effects	1.0	
Animation: Flipper	1.0	
Animation: Journeyman	1.47	
Animation: Multiplane	1.0	
Animation: Quick 2D	1.0	
Animation: Rotoscope	1.0	
Animation: Soundtrack	1.0	
Animation: Stand	1.0	
Animatrix: Modeler	1.21	
ARexx	1.20	
ARexx DB	2.0	
ART DEPARTMENT PRO	2.2.0 (AGA)	
Art Expression	1.0	
Asmone	1.0	
A-Sound Elite	1.0	
A-Talk III	1.0	
Audiomaster IV	1.0	
Audition 4	1.01	
Auto Cad Translator	2.10	
Autoscript	1.03	
Aztec C Developer	5.0b	
Aztec C Professional	5.0b	
Backup	3.5	
B.A.D.	4.13	
Bar Pro	3.0	
Bars & Pipes Pro	2.0	
Baud Bandit II	2.0	
Blitz Basic 2	2.0	
Boom Box	1.0	
Boot X	5.23	
Brilliance	1.0 (AGA)	
Broadcast Titler	2.0	
Butcher	2.0	
Byte 'N' Back	3.1.1	
C1-Text	3.1	
Caligari 24	1.0	
Caligari Broadcast	2.1	PAL
Caligari II	2.22	PAL
Can Do	2.0	
Cape 68k Assem	2.5	
Cell Pro	1.1	
Cinemorph	1.02 (AGA)	
Comeau C++	1.0	
Cross Dos Plus	5.01	
Cygnus Editor Pro	2.0	
DBman	5.0	
DELUXE MUSIC	2.0 (AGA)	
Deluxe Paint	4.5 (AGA)	
Deluxe Photolab	1.2	
Deluxe Video III	1.06	
Design Works	1.0	
DevPac	3.1	
Digi Deli View Gold	4.02	PAL
Digipaint	3.0	
Digital Sound Studio	1.15f	
Digi Works 3D	2.0	

PROGRAMMA RELEASE VERS.

DIRECTORY OPUS	4.03 (AGA)	
Diskmaster	2.03	
Disk Mechanic	2.7	
Disney Animation Studio	1.0	PAL
Distant Suns	4.2	
DKB Tracer	2.12	
Dos Two Dos	3.5	
Draw 4D PRO	1.0	
DynaCadd	2.04	PAL
Easy Amos	1.0	
Essence for Imagine Vol.1	1.0 FP	
Execellence	3.0	
Expert 4D	1.0	
Expert Draw	1.31e	
Fantavision	1.0	PAL
F-Basic	4.0	
F-Basic Source Level Debugger	4.0	
Final Copy II	2.0	
Fix Disk	1.2	
FlashBack	2.05	
Flexydump	2.0	
Flow	3.1	
Foundation	3.0	
Fractal Pro	5.1	
Genesis	1.10	
GFA Basic	3.52	
GFX Cad	3.1	
GIGA MEM	3.0	
Graphics WorkShop	1.01	
Ham Lab Plus	2.0.8	
Hard Disk Organizer	3.04	
HELM	1.0	
HighSpeed Pascal	1.0	
HiperCache Pro	1.0	
Hi Soft Basic	2.0	
Hot Links	1.1	
Hyperbook	1.0	
Image Finder	1.01	
Image F/X	1.03 (AGA)	
ImageMaster	9.23 (AGA)	
Imagine	2.0	PAL
Interchange Plus	2.0	
Interfont	1.0	
Interword	1.50	
JForth Professional	3.0	
KCS Level II	3.57	
Kick Pascal	2.1	
Kickstart	3.00	
Kindwords 3	1.0	
Lharc	1.21	
Lightwave 3D	2.0	
Lissa	1.2	
Lucypher	2.0	
Macro Paint	1.10	
Mac To Dos	1.1	
Mail-O-Dex-Professional	1.0	
Mandel Vroom	2.0	
Maple V	1.0	
Mathador	1.0	
MATH VISION	2.4	
Maxi Plan IV	4.09	
Maxi Plan Plus	2.0	
Medialink	3.0	
Midi Sample Wrench	2.0	
Migraph OCR	1.11b	
Minix	1.5	
MORPH PLUS	1.1.2 (AGA)	
Mostra	1.08	
Mr. Backup Pro	1.12	
Multitrace	1.0	
Neuro Pro	2.0	
NOTEBOOK	1.0	
Oktalizer	1.1	
Opticks	1.0	

PROGRAMMA RELEASE VERS.

Page Flipper Plus F/X	2.0	
Page Render	1.6	
PageSetter III	3.0	
Page Stream	2.21 HL	
Painter 3D	1.2	
PC Task	1.12	
Pen	PAL	1.4
Persist of Vision		
Ray Tracer	1.0u (AGA)	
Personal Fonts Maker	1.1	
PI IMAGE	3.1	
Pixel 3D	2.03	
Pixel 3D Professional	1.0	
Pixel Script	1.1	
Pixmate	1.1	
Pixound	2.5	
Plan It	4.0	
PoNGo (Morphus)	1.1	PAL
PowerPacker	4.3b	
Power Window	2.5	
Presentation Master	1.0	
Pro 24	1.0	
Pro Board Personal	3.0	
Professional Calc	1.4	
Professional Draw	3.02	
Professional Page	4.0 (AGA)	
Project D	2.0	
Pro Net Personal	2.0	
PROPER GRAMMAR	2.0	
ProText	5.5	
ProTracker	1.1a	
Pro Vector	2.1	
Pro Write	3.4 (AGA)	
Quadra 2D	1.3	
Quarterback	5.3	
Quarterback Tools	1.6	
Quick Pascal	1.40a	
Quickwrite	1.1	
Race Trace	1.32	
Raster Link	2.0	
Ray Dance	1.0	
Ray Shade	4.0	
Real 3D	1.4.2	
Reflections	2.0	
Resource	5.0	
Rexx Plus Compiler	1.2	
Sas/C Development System	6.0	
Saxon Publisher	1.2	
Saxon Script Pro	1.0	
Scala 500	1.0	
Scala CDTV	1.0	
Scala Info Channel	1.0	
Scala Multimedia	2.0	
Scala Videotitler	1.12	
Scope Maker	2.0	
Scene Generator	2.11	
Scenery Animator	2.06	
Sculpt 4D	2.09c	
Showanim	5.7	
Sisthema Plus	2.1	
SoundTracker	2.6	
Space Font Manager	1.0	
Spectracolor	1.0	
Stars F/X	1.1	
Stereo Master	1.0	
SuperBack	2.0	
SuperBase Professional IV	4.12	
SUPERJAMI	1.1	
Sybil	1.1	
SysInfo	3.01	
Take 2	2.0	
TERRAFORM	2.1 (AGA)	
Terrain	1.0	
The Director	2.0	
The Publisher	1.0	
The Publisher Color Pro	1.0	
The Texture Map Generator	1.0	
Thinker	2.1.4	
Touch up	1.03	
Transporter	1.1 (AGA)	
Transwrite	2.0	
True Print/24	1.0	
Turbo Imploder	4.0	
TurboPrint Professional	2.0	
Turbo Silver	3.01 SV	
TurboText	1.03	
Tv Paint (Arlequin)	1.7	

SEQUE ►

PROGRAMMA

RELEASE VERS.

Tv Paint (IV 24)	1.9
Tv show	2.0
TypeSmith	1.0
VDPaint	1.0
Vertex	1.73.1a
Video Director	1.0
Videoscape 3D	2.0
Video Sudio	3.0
VIEW	3.4 (AGA)
Virus X	4.40
Vista	1.2
Vista Make Path	1.0
Vista Pro	3.0H (AGA) PAL
Volum 4D jr.	3.4
Volum 4D Pro	3.2
VOYAGER	1.1
Waves	3.0
Will Winton's Playmation	1.47
Word Perfect	4.1
Wordworth	2.0
Workbench	3.0
Workbench Management Sys.	3.0
World Atlas	2.5
WShell	2.0
XCAD 2000	1.0
XCAD 3000	1.0
X-Cad 3D Pro	1.2a
X-Copy	3.3
Your Family Tree	2.2
Zoetrope	1.0

NOTE

Poche novità anche se mediamente interessanti. Iniziamo con la ASDG che promuove il suo prodotto di punta **Art Department Pro** alla versione **2.2**. Non si tratta di una versione molto migliorata rispetto alla precedente, ma può essere considerata una versione di affinamento e consolidamento in attesa dell'annunciata versione 3. L'impostazione dei selettori è ora identica a quella di **Morph Plus**. Modifiche anche per **Morph Plus** (versione 1.1.2) della stessa casa, pacchetto per il morphing e operazioni di image processing in animazione. Sono state inserite modifiche nel salvataggio dei file IFF, due nuove modalità di dithering, possibilità di scelta dell'ammontare del dithering, un selettore per la scelta veloce del modo grafico Super72, possibilità d'apertura per gli operatori Visual, Perspective, Ripple, Rotate, Sphere, Twirl, Warp di uno schermo in 8 bitplane in toni di grigio per macchine con AGA, aggiunti altri comandi **ARexx**; anche **FRED** (l'utilità acclusa per l'esecuzione di operazioni di image processing automatiche su sequenze di frame) si avvantaggia ora del nuovo chipset. L'utilità **PD View** è stata considerata in un recente sondaggio condotto nella mailing list dedicata a *Imagine* di Steve Worley, il miglior player di file Anim in assoluto. Ne viene ora rilasciata una splendida versione 3.4 dotata di help in linea e in grado di eseguire fino a 60 frame/sec. Il programma visualizza animazioni in risoluzione e numero di colori introdotti col chipset AGA, con pieno supporto anche dei 262 mila colori dell'HAM8. **Directory Opus** della Inovatronics è stato rilasciato in versione 4.03. A causa dei problemi nel caricamento dei formati grafici nei modi non standard Sham/DymanicHyres e il fatto che questi sono stati resi obsoleti dal nuovo chipset AGA, tali formati sono stati definitivamente rimossi. La stessa Commodore aveva imposto ai programmatori registrati di non supportarli più. Il modulo di copia ora possiede un selettore supplementare che consente di eseguire una comparazione traccia-traccia tra due dischetti per controllarne un eventuale contenuto identico. **Math Vision 2.4** della Seven Seas Software aggiunge nuove funzioni matematiche alle precedenti, **ARexx** e ora supporta direttamente le schede video Video Toaster e IV24. **AMPLIT 2.0** della SciTech Software è un avanzato programma per il disegno professionale di grafici. Molto semplice da utilizzare, il programma possiede un'interfaccia **ARexx** e supporta i formati EPSF, IFFDR2D, PostScript e HPGL. La Eagle Tree Software, lavorando sull'upgrade del glorioso **Butcher** (apprezzato dagli omigisti più anziani per la sua facilità e versatilità), ha preparato un nuovo programma chiamato **Helm**, un pacchetto orientato al multimedia con un occhio particolare alla manipolazione complessa delle immagini. **A.D.L.**

ENTITY

Una protagonista femmina
nel mondo dei videogiochi!

Computer: Amiga
Supporto: Disco
Prezzo: n/d
Produzione: Loricel

GIUDIZIO
COMPLESSIVO:
BUONO



Grafica:	★★★★★
Sonoro:	★★★★★
Giocabilità:	★★★★★
Prezzo:	★★★★★

L'evento è di quelli di portata storica: per la prima volta in un videogioco d'azione la protagonista è una ventenne nel pieno fulgore della sua giovinezza, con jeans attillati e forme procaci. Che si fa? Si mandano i bambini a letto? No, si festeggia finalmente l'ingresso della donna a pieno merito nell'olimpo dei personaggi videoludici. Tanto più che la ragazza



sembra avere una determinazione mascolina nell'affrontare le insidie: nonostante le si facciano incontro mostri di ogni genere, la fanciulla non si scompone e anzi risponde con il potenziale (quello bellico...) di cui dispone.

Come qualcuno avrà capito, *Entity* è un tipico platform con una dose non indifferente di pestaggi, paragonabile a uno dei tre capitoli di *Shadow of the Beast*. In più, però, i programmatori francesi hanno pensato d'inserirci qualche intermezzo avventuriero con enigmi, che fanno di *Entity* qualcosa di un po' più profondo di un semplice gioco di violenza a scorrimento. Così come in *Beast 3*, insomma, bisogna fare anche un certo esercizio di ragionamento per superare alcune fasi del gioco, manipolando oggetti per innescare particolari meccanismi. La grafi-

ca è veramente di buon livello, con una forte parvenza di realismo data soprattutto da certi dettagli riguardanti i mostri. Molto bello anche l'effetto sfumato del cielo mentre la musica dà il suo tocco di atmosfera al tutto.

Nel complesso, *Entity* si dimostra un gioco molto immediato, con un metodo di controllo tranquillamente manovrabile e con la possibilità di fuoco continuo che si è dimostrata una scelta azzeccata.

P.C.

ABANDONED PLACES 2

Non solo sotterranei

Computer: Amiga
Supporto: Disco
Prezzo: L. 99.900
Produzione: I.C.E.
Distribuzione: Leader (Via Adua 22, 21045 Gazzada Schianno - ☎ 0332/874111)

GIUDIZIO
COMPLESSIVO:
BUONO



Grafica:	★★★★★
Sonoro:	★★★★★
Giocabilità:	★★★★★
Prezzo:	★★★★★

Dopo il passaggio di un gioco come *Legends of Valour*, che proponeva per la prima volta una grafica con texture mapping, possiamo aspettarci prossimamente grandi cose. Per adesso, accontentiamoci di proseguire la saga dei giochi di ruolo in soggettiva: *Abandoned Places 2* è ancora una volta un esempio classico. Dopo la prima puntata, vi tocca ancora salvare il regno di Kalynthia, nuovamente tempestato di presenze malvage. Prima di partire all'avventura, avrete bisogno di allestire il consueto "party" di quattro soggetti e dopo aver scelto tra i 32 aspiranti eroi, sarete pronti per affrontare i 35 livelli di gioco. Il gioco inizia come al solito nei "dungeon" affollati d'insidie, ma per fortuna vi è un tratto in cui si possono abbandonare le segrete: vi aspettano campi e foreste, dopodiché si tornerà "sotto". Più si va avanti, più ovviamente si

guadagnano punti-esperienza e si ottengono nuovi incantesimi per mago e chierico; all'interno dei dungeon non mancano naturalmente i pulsanti da premere per aprire passaggi segreti, oggetti da infilare da qualche parte per innescare particolari meccanismi. L'interfaccia è abbastanza classica ed è comoda da gestire a parte qualche problema con i due tasti che a volte assolvono due funzioni diverse e se si



sbaglia mentre si è in battaglia sono davvero guai seri. A parte questo, *Abandoned Places 2* è un buon prodotto: la grafica in half-brite è ottima e i mostri hanno quel *quid* di tridimensionale. La difficoltà è progressiva e il gioco non manca degli ingredienti giusti a farne oggetto d'acquisto da parte di tutti i giocatori di ruolo.

P.C.

PRIME MOVER

Su due ruote con la *Psygnosis*

Computer: Amiga

Supporto: Disco

Prezzo: n/d

Produzione: Psygnosis

Distribuzione: C.T.O. (Via Piemonte 7/F, Zola Predosa - ☎ 051/753133)

GIUDIZIO
COMPLESSIVO:
DISCRETO



Grafica:	★ ★ ★ ★ ★
Sonoro:	★ ★ ★ ★
Giocabilità:	★ ★ ★ ★
Prezzo:	

P *Prime Mover* è ambientato nel mondo delle due ruote a motore. L'ultimo grande gioco motociclistico su Amiga è stato *No Second Prize* della Thalion, che manteneva ancora l'ambientazione poligonale, così come *Red*

LUDO NEWS

Le novità dell'ECTS
di Londra

a cura di Paolo Cardillo

G ià la tendenza si stava intravedendo, ma con Londra se n'è avuta l'assoluta conferma. La presenza delle console giapponesi sta pesantemente influenzando l'orientamento delle case produttrici di software, che si sono buttate a capofitto nella produzione di cartucce per i "sistemi senza tastiera" del Sol levante. Ma questo non è necessariamente un male: la produzione per Amiga sta sì calando, ma, vista la concorrenza delle console, i nuovi giochi sono realizzati con molta più cura e basta dare un'occhiata agli ultimi titoli per accorgersi dell'incremento generale della qualità.

Solo due parole per gli ECTS Awards, i premi della fiera londinese: *Monkey Island 2* ha vinto nelle categorie RPG/Avventura e Colonna sonora, *Formula 1 Grand Prix* come Simulazione, e *Indiana Jones & the Fate of Atlantis* è stato giudicato il miglior gioco per computer. La Electronic Arts è invece la software house dell'anno.

Ora vediamo le novità. Non che il CD-ROM abbia fatto proprio furore in questa edizione dell'ECTS, ma qualcosa è stato annunciato: probabilmente il progetto più ambizioso è quello della Storm, che ha intenzione di portare su dischetto d'argento *Lawnmower Man 2*, vale a dire il seguito del primo film sulla realtà virtuale (in Italia, // *Tagliaerba*). Dopo il clamoroso fallimento del progetto di trasposizione del primo film, funzionerà anche su CDTV. Teoricamente, dovrebbe essere qualcosa di stupefacente: alla Storm hanno annunciato 30 livelli con dieci diversi stili di gioco e tanta, ma veramente tanta grafica strabiliante d'intermezzo. Non per nulla il gioco vedrà la luce solo nel 1994. Nel frattempo, potremo consolarci con qualcosa di più classico come *Total Carnage*, della ICE. Questo gioco è il seguito di *Smash TV*, vale a dire uno dei giochi con la maggior concentrazione di sprite in un solo schermo della storia: chiusi in una stanza si trattava di farli fuori tutti, partecipando a una specie di gioco a premi ultraviolento. *Total Carnage* avrà invece uno sfondo bellico e ancora più distruzione di massa. Sempre della ICE, mi ha fatto una discreta impressione *Gulp the Suppy*, un gioco con protagonista un pesciolino rosso davvero simpatico.

Per gli amanti dello sport, la Domark presentava

Championship Manager '93, versione riveduta e corretta di quel *Championship Manager* talmente carico di dati da dover aspettare ore per vedere i risultati delle partite di calcio. Stavolta ci hanno assicurato che non ci saranno problemi di calcoli, anzi il gioco avrà 50 nuove caratteristiche in più. *Formula 1 Champions* sarà invece un gioco arcade a quattro ruote programmato dagli stessi autori dell'apprezzatissimo *Vroom*. Ma il programma che più incuriosiva era il **Flight-Sim Toolkit**, che come qualcuno avrà capito non è null'altro che un creatore di simulatori di volo. Si disegna il paesaggio, si crea l'aereo, si concepisce l'abitacolo con la strumentazione e tutto è pronto. Be', visti i trascorsi della Domark con i Kit per il 3D non so quanto ci sia da fidarsi: staremo a vedere. Una notizia importante: la Domark ha ora sotto contratto la Maelstrom, cioè il team capitanato da Mike Singleton, che come qualcuno saprà è l'autore di grandi giochi come *Midwinter 1* e *2* e del recente *Ashes of Empire*. Una capatina alla Renegade è stata l'occasione per una chiacchieratina con Andrew Braybrook, uno dei più famosi programmatori di videogiochi, che sta dando gli ultimi ritocchi al suo **Uridium 2**. Alcuni dati: 48 frame di animazione per l'astronave (la "Manta"), 32 colori, scorrimento di 50 frame/sec. e due giocatori contemporaneamente. Stranamente, l'intero gioco è in NTSC: Andrew ha spiegato che l'area di gioco sarebbe stata troppo grossa per quello che aveva in mente, e così ha apportato questa modifica. Lo sparatutto si è comunque dimostrato veloce e parecchio divertente nel modo a due giocatori. Altro gioco targato Renegade è **Ruff & Tumble**, ottimo platform con una grafica notevole. Alla Gremlin cominciano a fare il nome di **Zool 2** senza però rivelare niente. Di **Lilil Divil** si è potuto invece vedere abbastanza da confermare l'ipotesi che questa sorta di avventura grafica arcade, con protagonista un simpaticissimo demone, potrebbe sfondare.

Infine, veniva presentato **Hero Quest 2**, seguito del gioco da tavolo diventato videogioco: stavolta abbiamo dieci scenari con stili differenti, quattro nuovi personaggi, quattro giocatori in contemporanea e altro ancora. La grafica è migliorata ed è un appuntamento che i "nuolisti" non dovrebbero perdere. La Microprose, poi, ha finalmente annunciato l'uscita di **A.T.A.C.** sull'Amiga: si tratta di una simulazione di volo in cui piloterete sia un elicottero, sia un caccia, dotati entrambi di armamento sofisticatissimo. Il pretesto per decollare è la presenza di spacciatori di droga anch'essi dotati di un arsenale non indifferente. Da segnalare anche l'uscita di una sorta di arcade militaresco che avrà come protagonisti Napoleone, Wellington e Blucher. Si chiama **Fields of Glory - The Road to Waterloo** e non sembra niente male. □

Zone. Adesso però la Psygnosis ha deciso di ripiegare sugli sprite e così infatti vengono rappresentati tutti i concorrenti in gara. La pista è molto larga e lo scorrimento in generale è piuttosto fluido, il che dona una buona sensazione di velocità, grafica laterale compresa. Come prima impressione, *Prime Mover* lascia favorevolmente colpiti: a parte le varie opzioni di campionato, è proprio la celerità dell'asfalto che vi scorre sotto le ruote che "carica". Tra l'altro, gli sprite sono ben ritratti e insomma il look predispone a giocarci. Una volta che s'iniziano a sorpassare i vari concorrenti, si scopre che anche *Prime Mover* non è però esente da un errore abbastanza classi-

co che si verifica in vari giochi di guida: si affrontano spesso tratti lunghi senza che s'incontri l'ombra di un avversario sulla pista. In questo modo il gioco sarà anche realistico, ma non mi dicano che avevano intenzione di fare qualcosa di simulativo: *Prime Mover* doveva essere un gioco con un buono spirito arcade, e sarebbe stato sufficiente includere qualche centauro avversario in più da sorpassare giusto per mantenere viva l'attenzione del giocatore. Si tratta quindi di un prodotto ben realizzato, ma che manca della necessaria qualità arcade che mantenga abbastanza desto il giocatore.

P.C. ■

SUPERGAMES

Via Vitruvio n. 37 - 20124 Milano
Tel. e fax 02/29520184-29520180

ORARI APERTURA: 9.00 - 12.30/15.00 - 19.30

DA 12 ANNI NEL SETTORE - LA QUALITÀ INFORMATICA TOTALE NELL'HOME E PERSONAL COMPUTER

AMIGA 4000

The Super Amiga
68040 25 MHz - Kick 3.0
2 MB CHIP - 4 MB FAST
ChipSet AGA 32 Bit
Hard Disk IDE 120MB
L. 3.990.000
Con HD 220MB
L. 4.400.000
SIMM 4MB 32 Bit
L. 350.000

AMIGA 4000

68030 - 4MB - 85MB HD
L. 2.650.000

AMIGA 1200

The New Generation
68020 14 MHz - Kick 3.0
2 MB CHIP 32 Bit
L. 790.000

Con HD IDE 40MB
L. 1.190.000

Con HD IDE 80MB
L. 1.450.000

COMMODORE CDTV

Multimedia Computer
Con drive, tastiera, mouse
Enciclopedia Grolier su CD
L. 900.000

In arrivo dagli USA la nuova raccolta
di software di pubblico dominio

EXECLIB

Per AMIGA con sistema operativo
2.0 o superiore
Telefonare per informazioni

PC 1204

4MB
L. 489.000
4MB 40 MHz
L. 890.000

AMIGA COMPUTERS

A500 v 1.3 512K L. 490.000
A600.....L. 499.000
A600HD 40MB L. 890.000
A2000.....L. 850.000

Monitor 1084S L. 430.000
Monitor 1960.....L. 650.000

PCMCIA RAM CARD

Fast RAM
per A600/A1200
2MB Card L. 290.000
4MB Card L. 490.000

MICROBOTICS MBX 1200

FPU Accelerator Card + Clock
68881 14MHz 0MB ..L. 379.000
68881 14 MHz 4MB ..L. 750.000
68881 14MHz 8MB ..L. 890.000
68882 25MHz 0MB ..L. 529.000
68882 25MHz 4MB ..L. 950.000
68882 25 MHz 8MB ..L. 1.150.000

ACCESSORI AMIGA

512K RAM x A500 L. 59.000
1MB RAM x A500 Plus L. 150.000
1MB RAM x A600 Originale...L. 130.000
KICKSTART 1.3 x A600 L. 69.000
KICKSTART 1.3 x A500 L. 69.000
DRIVE ESTERNI da L. 120.000
HARD DISK 33MB x A500 L. 490.000
GVP 80MB Hard Disk x A500 ..L. 950.000
NEXUS SCSI CARD x A2000 ..L. 350.000
A570 CD-ROM x A500.....L. 550.000
SCANNER ALFADATA da.....L. 249.000
Genlock GVP G-LOCK L. 1.150.000
RocGen Plus.....L. 590.000
Video Backup System L. 69.000
Amiga Network System L. 69.000

HARD DISK PER A600 - A1200

40MB IDE 2,5"
L. 430.000
60MB IDE 2,5"
L. 500.000
80MB IDE 2,5"
L. 650.000
120MB IDE 2,5"
L. 950.000

TUTTI I PREZZI
SONO IVA COMPRESA

DRIVE ESTERNO
HIGH-DENSITY
1.76 MB
PER AMIGA
Telefonare

OGGI VENDERE UN BUON PRODOTTO NON BASTA!

NOI VI OFFRIAMO UN'ASSISTENZA POSTVENDITA HARDWARE E SOFTWARE
ASSIEME AL PRODOTTO CHE ACQUISTATE, UN SERVIZIO IN PIU' PER CHI DESIDERA
QUALITA' E SERIETA', NON SOLO PAROLE.

VENITE A TROVARCI, VERRETE ACCOLTI DA PERSONALE QUALIFICATO CHE VI
AIUTERA' NELLA SCELTA DEL SISTEMA PIU' ADATTO ALLE VOSTRE ESIGENZE.

SALONI D'EUROPA

TUTTE LE NOVITÀ DEL CEBIT '93

***Dai nuovi PC Commodore all'A4000 Tower,
nuove ed entusiasmanti soluzioni grafiche per Amiga,
il DCC della Philips contro il MiniDisc della Sony...***

di Marco Dufour

Come ogni anno, si è tenuto ad Hannover il CeBIT, e come ogni anno il successo dell'esposizione fieristica è stato grande. Il CeBIT è la più importante manifestazione internazionale dedicata al mondo dei computer, con i suoi 5402 espositori e una superficie fieristica pari a 320 mila metri quadri.

Il centro fieristico di Hannover si è nuovamente proposto come piattaforma unica nel suo genere a livello internazionale nel settore dei prodotti per l'ufficio e delle tecnologie dell'informazione e della telecomunicazione. Con questa fortunata edizione è stata superata la soglia del mezzo milione di visitatori (660 mila), giunti negli otto giorni della fiera per informarsi sull'offerta internazionale del settore e sulle nuove tendenze che in esso si stanno delineando. L'offerta del CeBIT '93 ha occupato complessivamente tutti i 22 padiglioni della fiera di Hannover, chiaramente divisa per gruppi tematici, spaziando dai prodotti classici per l'ufficio e dall'elaborazione dati alla produzione computerizzata, alle tecniche bancarie e ai sistemi di sicurezza, per arrivare ai settori della telecomunicazione e delle reti.

Anche quest'anno il CeBIT ha reso onore alla sua fama di palcoscenico internazionale delle novità. Nel programma espositivo del '93 si è rivelato particolarmente interessante in questo senso il settore del software, peraltro notevolmente ampliatosi, e i quattro padiglioni dedicati al tema delle telecomunicazioni. È risultato notevolmente

allargato anche il settore delle reti che ha concentrato le sue applicazioni più attuali all'interno di due enormi padiglioni. Per la prima volta, entrambi i padiglioni erano dotati di una rete radio LAN a installazione fissa, che ha consentito di dimostrare le possibilità dei diversi componenti attraverso l'impiego diretto sul posto.

Mancare al CeBIT significa perdere l'occasione più interessante per seguire veramente da vicino il mercato dei

trovarsi all'ingresso di uno stadio calcistico. Una fiumana di gente, fra cui molti stranieri, convogliava dalla fermata della metropolitana ai cancelli della fiera. In qualsiasi angolo della città erano visibili cartelli e indicazioni per raggiungere la fiera; bisogna anche dire che tutta la città di Hannover vive grazie al turismo portato dalle sue diverse esposizioni annuali per cui risulta molto naturale tutta questa organizzazione.

I diversi padiglioni erano appunto suddivisi per argomenti e tipo di espositori. Nel padiglione numero 1 erano per esempio presenti gli stand delle principali società leader del settore informatico; si andava dall'IBM alla Nec, dalla Hewlett Packard alla Commodore.

Lo stand della Commodore

Ebbene sì, la Commodore era proprio tra i grandi, occupando uno stand di notevoli dimensioni, decisamente attraente. C'è da dire che in Germania la considerazione per i computer Commodore è davvero alta, sia per il mercato Amiga che per il mercato PC. Lo stand era composto da un corpo principale lungo una cinquantina di metri, a due piani: nel piano terra erano ospitate aziende con prodotti dedicati ai modelli Amiga e PC, mentre al piano superiore venivano stipulati contratti commerciali e rilasciate spiegazioni particolareggiate dei nuovi prodotti presentati.

La Commodore presentava tutta la nuova linea PC, sensibilmente rinnovata e decisamente appetibile. Veniva final-



L'immenso complesso fieristico in una veduta dall'alto

computer, Commodore e non. È anche vero però che essendo tantissimi gli espositori ed essendo tantissime le novità, è estremamente complesso riuscire a trarre in poco tempo delle conclusioni su quale sia la migliore soluzione hardware/software o il miglior sistema di sviluppo.

La fiera

Appena entrati nel complesso espositivo la prima sensazione è quella di

mente presentato il nuovo 486 SLC a 25 MHz, un potente PC a 32 bit caratterizzato da un prezzo contenutissimo (circa 1.300.000 lire) reso possibile dall'utilizzo della nuova versione economica del processore Intel. Questo modello viene a costituire l'entry level della linea PC Commodore, a pari prezzo con il 386DX a 33 MHz: il 486 SLC è un processore a basso consumo, destinato prevalentemente ai portatili, al quale manca il coprocessore matematico. Ne risulta un computer di discreta velocità, ma compatibile con gli indirizzi di comandi di tutta la serie 486.

Al già famoso PC 486 50C è stato affiancato l'ultraveloce 486 a 66 MHz in standard DX2. In Germania viene venduto a 3500 marchi (circa 3.400.000 lire) nella configurazione con hard disk da 120 MB, 4 MB di RAM (espandibili su scheda fino a 32), scheda video SuperVGA da 1 MB e sette slot di espansione a 16 bit. Il prezzo sembra abbastanza ragionevole, senza tener conto che ormai Commodore è un marchio famoso e non va confuso o paragonato con i "taiwanesi assemblati". Tutta la serie PC ha comunque visto una rielaborazione globale per cui è possibile che per un certo periodo anche in Italia possa esserci un po' di confusione su quali siano i nuovi o i vecchi modelli.

Per quanto riguarda la linea Amiga, tutto l'interesse dei professionisti era rivolto all'Amiga 4000 e alle sue nuove schede. Finalmente, ha fatto la sua comparsa la tanto sospirata scheda A4091, un controller in standard SCSI2 a 32 bit per Amiga 3000 e 4000. Di questo controller si parlava ormai da mesi, più esattamente da quando la Commodore aveva presentato nel settembre 1992 la nuova serie Amiga con i chip AGA. In quella occasione, a risposta delle numerose critiche da parte della stampa mondiale sulla mancata implementazione di un controller SCSI sui nuovi modelli Amiga, la Commodore annunciò "l'imminente" uscita di questa scheda. Quella che si poteva ammirare all'interno di un Amiga 4000 "spaccato" era comunque un prototipo, ma il fatto che fosse già disponibile un manuale con tutte le caratteristiche induce a pensare che la reale disponibilità di questa scheda non dovrebbe subire troppi ritardi. I programmi dimostrativi eseguiti esaltavano la sua velocità, e il pannello con le caratteristiche lo citava come il controller SCSI2 più veloce sul mercato. Bisognerà vedere come si comporterà la versione definitiva che comunque si aggirerà attorno ai 10 MB di dati trasferiti al secondo, una velocità decisamente interessante.

Vicinissimo al controller A4091 era in bella mostra un prototipo di Amiga 4000 Tower. Anche questo modello era atteso da tempo, soprattutto da quando sulle riviste di tutto il mondo cominciarono a circolare le novità promesse alla Devcon svoltasi a Orlano (USA) lo scorso gennaio. Le notizie trapelate parlavano di un imminente Amiga 4000 Tower con tanto di controller SCSI2 e DSP (Digital Signal Processor). Notizie che non ricevettero smentita, ma solo critiche da parte di alcuni osservatori all'interno della Commodore, che non vedevano di buon occhio il fatto di presentare le caratteristiche di prodotti futuri, non ancora commercializzati. Nonostante questo, a distanza di pochi mesi viene presentato anche il modello Tower, con processore Motorola 68040 a 25 MHz su scheda separata (per poter inserire la futura versione di processore, il Motorola 68060), 5 slot a 32 bit Amiga e 4 PC, hard disk da 100 o 200 MB e 6 MB di RAM espandibili su scheda a 18. La novità più interessante è comunque l'implementazione su questo modello della tecnologia SCSI2, caratterizzata da un'altissima velocità di trasferimento dati. Dopo un'accurata analisi, si è potuto notare che il controller SCSI2 è montato direttamente su scheda, lasciando libero un ulteriore slot a 32 bit.

La velocità dell'Amiga 4000 Tower veniva dimostrata grazie a un'animazione a 256 colori creata con il programma di generazione frattale *VistaPro*. L'animazione della durata di qualche minuto veniva caricata direttamente dall'hard disk e spedita a grande velocità al processore grafico: il risultato era interessante, ma i piccoli scatti che ogni tanto venivano a formarsi sull'animazione non dimostravano certamente una qualità perfetta: è forse una pignoleria eccessiva, ma già che c'erano potevano ottimizzare l'animazione e conferire una migliore riproduzione. Del tanto acclamato DSP invece non veniva fatta menzione, e di conseguenza si possono trarre due conclusioni: o l'Amiga 4000 Tower presentato al CeBIT '93 non era altro che un prototipo allo stadio primordiale (ne conseguirebbe una commercializzazione non prima di luglio) o il DSP non sarà inglobato nella serie Amiga ancora per lungo tempo.

Di prossima commercializzazione, invece, è la nuova serie di monitor per Amiga, il 1940 e il 1942. Si tratta di due monitor con le stesse caratteristiche tecniche a eccezione del dot pitch (la grandezza del puntino luminoso del video, il pixel o picture element): il primo è un .39 destinato al mercato consumer con un ottimo rapporto prez-

zo/prestazioni, il secondo è un .28, destinato al mercato professionale. La cosa veramente interessante è il funzionamento BiSync, che permette la visualizzazione di quasi tutte le risoluzioni grafiche (comprese quelle deinterlacciate dei chip AGA) tranne il modo Super 72 (800 x 600). Le frequenze su cui lavorano questi monitor vanno da 15.6 a 15.8 KHz e da 27.3 a 31.5 KHz. La frequenza verticale varia da 47 a 75 Hz.

Se qualcuno di voi sta pensando alla morte del monitor multisync 1960 è meglio che si ricreda. Come già visto, le frequenze sono diverse, e la differenza sostanziale tra questi modelli è la presenza o meno di speaker audio stereofonici e la qualità o meno delle immagini visualizzabili. Se da un lato il 1942 offre la soluzione ideale per chi possiede un Amiga 1200, può risultare più professionale rimanere sul "vecchio" 1960 se si hanno intenzioni più serie. Faceva bella mostra anche un 1084S nero, dedicato al CDTV e alle applicazioni multimedia. Non si sa ancora se questo modello verrà commercializzato anche in Italia in quanto l'unica differenza rispetto al normale 1084S è la colorazione.

Dentro la Commodore

All'interno del grande stand Commodore erano ospitati diversi sviluppatori di prodotti dedicati alla serie Amiga e PC. Grandi nomi tedeschi come Maxon, Boeder, Electronic Design e altri presentavano diverse novità. Questa edizione del CeBIT sembrava dedicata alla grafica in generale: tutte le società ospiti presentavano diverse schede grafiche e prodotti dedicati alla manipolazione delle immagini.

Prima di addentrarmi nelle novità, vorrei restituirvi questa suggestiva immagine davanti alla quale mi sono trovato: in un lato dello stand, gremito di gente, c'era un videowall di dimensioni mastodontiche: la folla guardava esterefatta le animazioni create con l'Amiga grazie al programma *Morph Plus* (giunto ora alla versione 1.2). Devo ammettere che ero già stato ben impressionato da questo programma, ma vedere delle animazioni di quelle dimensioni mi ha lasciato veramente a bocca aperta.

Sullo schermo gigante funzionavano ininterrottamente le presentazioni dei diversi programmi e delle diverse soluzioni hardware/software dedicate all'Amiga.

A pochi mesi dalla sua reale commercializzazione, è stata presentata la nuovissima versione 1.2.0 di *Morph Plus*. Per chi non avesse mai sentito parlare di questo famosissimo programma (recen-

sito sul numero di novembre '92), ne ricordo le caratteristiche principali: è prodotto dalla ASDG (925 Stewart St., Madison, WI 53713, USA), ha una struttura modulare del tutto simile ad *Art Department Professional* (sempre della ASDG) ed è in grado di manipolare immagini e animazioni a 24 bit (16 milioni di colori), creando anche metamorfosi tra un'immagine e l'altra. Con la nuova versione, sono stati aggiornati tutti gli operatori per gestire al meglio le risoluzioni AGA e la modularità grafica del sistema operativo 3.0.

Gli operatori detti "Visual" ora visualizzano schermate a 8 bit (256 colori) e si possono espandere fino alle super risoluzioni interlacciate (1280 per 512 pixel). Finalmente, è possibile visualizzare immagini in risoluzioni non interlacciate grazie al comando DFLT ed è stato aggiunto un operatore detto "Sim Print" per la simulazione della resa finale di stampa. Più che una rivoluzionaria ridefinizione del programma, sembra molto un perfezionamento di quello originale che con i nuovi Amiga (4000 e 1200) aveva qualche problema.

Nuove soluzioni grafiche

Come già accennato, le società che presentavano soluzioni hardware per la gestione della grafica erano molteplici. Tra queste spiccava la Village Tronic (Braunstrasse 14, 3000 Hannover, Germania), che presentava la strabiliante scheda grafica Picasso 2. Questa scheda in standard Zorro 2 (funziona anche su Amiga 2000) è in grado di raggiungere risoluzioni fino a 1280 per 1024 punti in 16 milioni di colori. Molto interessante è il fatto che la scheda viene riconosciuta come risorsa di sistema e può essere utilizzata da qualsiasi programma che faccia uso dello schermo *Workbench*. Sotto sistema operativo 3.0 è in grado di visualizzare un *Workbench* a 256 colori anche sull'Amiga 2000, esattamente come i chip AGA della Commodore. Chi si aspettava che questo sarebbe successo, è stato decisamente accontentato.

tato.

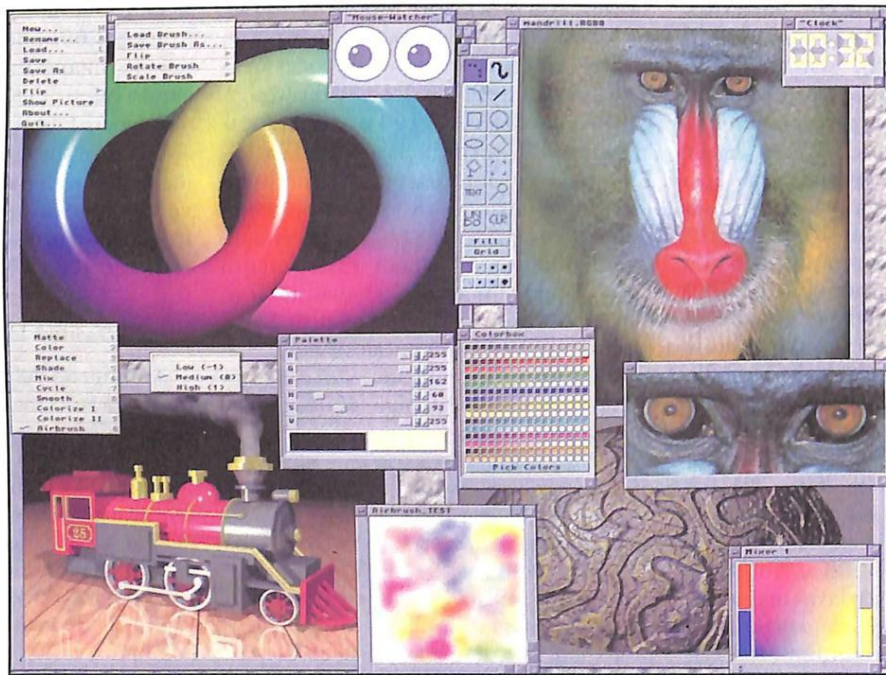
Con un buffer video da 1024K e la possibilità di trasferire i dati al Blitter alla strabiliante velocità di 30 MB al secondo, questa scheda rappresenta la soluzione ideale per chi avesse un "vecchio" Amiga 3000 e volesse sfruttare appieno tutte le caratteristiche delle nuove modalità grafiche. Assieme alla scheda vengono forniti diversi programmi tra i quali alcuni per la decodifica dei file IFF24, Gif e Jpeg, nonché un piccolo player per animazioni Mpeg (lo standard Mpeg serve per la compressione/decompressione del full motion video; lo si può impiegare anche per sequenze video compresse su hard disk).

Il suo prezzo, solo 598 marchi (circa

contemporaneamente, di cui uno visualizzava un *Workbench* normale a 16 colori, un altro un'immagine a 24 bit e un altro ancora un super *Workbench* a 24 bit di dimensioni spropositate. Quello che potete vedere nell'immagine riportata in questa pagina è sensazionale: pensate che chi dimostrava il prodotto, dopo aver velocemente disegnato una di queste figure, si è messo a giocare sulla sua posizione, spostando le finestre da un monitor all'altro semplicemente portandole a destra o a sinistra dello schermo. Ciò che era maggiormente interessante era il fatto che l'immagine, passando attraverso lo schermo *Workbench* a 16 colori si retinava in tempo reale, a una velocità impressionante. Questa scheda in standard Zorro 3 ha 4 MB di RAM video e una risoluzione massima fino a 1600 per 1280 punti in 16 milioni di colori.

Maxon Computer ed Electronic Design

Chi legge le riviste tedesche dedicate al mondo Amiga, avrà certamente notato questa software house, che è decisamente proliferata. La Maxon Computer (Postfach 5969, w-6236 Eschborn, Germany) presentava *Maxo-*



L'incredibile interfaccia della scheda Rainbow 3, la prima con il nuovo standard EGS

560 mila lire), contribuirà a un sicuro successo.

Sempre all'interno dello stand Commodore, esponeva la società Ingenieurbüro Helfrich (Am Wollager 8, 2870 Delmenhorst, Germany) che presentava in anteprima mondiale la nuova scheda grafica Rainbow 3. Questa scheda è un vero e proprio coprocessore grafico a 24 bit, in grado di accelerare le operazioni di disegno, visualizzazione e ritocco delle immagini digitali. Forse qualcuno ha già sentito parlare di EGS (Extended Graphic System), un sistema di visualizzazione grafica messo a punto da un gruppo di società tra cui anche la GVP. Questa è stata la prima volta che una scheda di questo tipo veniva presentata al pubblico in versione commerciale.

Un Amiga 4000 pilotava tre monitor

nWord, *MaconCad2*, *MaxonPaint*, *MaxonC++* e *FastRay* per Amiga, oltre ad alcune versioni simili per *Windows*. Il commento è molto positivo, i programmi sono molto ben fatti, l'unico problema è che sono prodotti esclusivamente in lingua tedesca. Alle ripetute domande su una possibile versione inglese o addirittura italiana, è stato risposto che "non è nei loro interessi un tale lavoro" e che le uniche localizzazioni saranno possibili solo con la *Locale.library* del nuovo sistema operativo. Un discorso che non regge molto e che purtroppo mi ha dato una visione un po' negativa della politica commerciale della società. I programmi da loro prodotti avrebbero un discreto successo anche in Italia, se almeno fossero scritti anche in inglese. Se conoscete il tedesco non esitate a

rivolgerli alle soluzioni Maxon, altrimenti provate a convincerli che converrebbe anche a loro espandere il proprio mercato commerciale.

La Electronic Design (D-8000 Muenchen, 45 Detmoldstrasse 2a, Germany) presentava diverse soluzioni di genlock, dall'amatoriale al professionale di qualità broadcast. All'interno dello stand venivano esposte alcune periferiche molto interessanti, e mostrato un video interamente realizzato con l'Amiga: questa videocassetta in formato Pal VHS è stata chiamata (con poca fantasia) *Take Two* ed è indirizzata agli utenti poco esperti di computer, ma con grandi necessità nel campo della post-produzione video. All'interno del video vengono descritte le caratteristiche di ogni modello di apparecchio prodotto dalla Electronic Design e alcuni trucchi per raggiungere effetti speciali sorprendenti con apparecchiature molto semplici.

Il primo modello amatoriale è il Pal-Genlock, con ingressi Pal videocomposito e la possibilità di separare il segnale RGB per i digitalizzatori che ne hanno bisogno. Il segnale non è certo di qualità broadcast, ma per chi è alle prime armi con attrezzature video è già un buon inizio.

Più professionale è invece il Genlock Y/C, molto simile come caratteristiche al primo, ma in grado di supportare il segnale Super video. Può essere utile anche per la digitalizzazione in altissime risoluzioni o per la conversione del formato Y-C per l'utilizzo su apparecchiature RGB. Decisamente interessante è invece la nuova scheda Frame-Machine in standard Zorro 2 (funziona anche sull'Amiga 2000), che è in grado di digitalizzare in tempo reale anche sequenze d'immagini a 24 bit, permettendo anche l'assemblaggio di diverse sequenze. Con la scheda FM-Prism24 (da aggiungere a Frame-Machine) è possibile visualizzare i 16 milioni di colori. Questa potente scheda, se collegata a un genlock è in grado di creare il Picture In Picture in qualsiasi dimensione, creare dissolvenze complesse tra sequenze video, tendine e altre diavolerie del genere. Il segnale video sembrava di buona qualità e gli effetti attribuiti alle sequenze video erano decisamente superiori alla media.

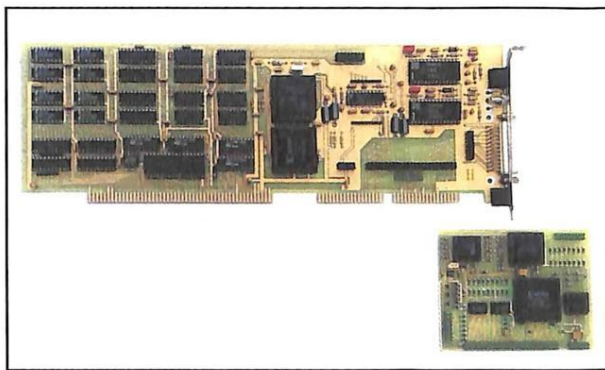
Facevano bella presenza all'interno del CeBIT anche altri espositori di prodotti per Amiga. Tra questi ricordo la Vortex, che presentava la nuova scheda 486 SLC per Amiga 2000/3000/4000. Sempre fuori dalla Com-

modore abbiamo incontrato i simpatici standisti della Rombo (i cui prodotti vengono importati dalla Soundware di Varese, Tel. 0332/232670), che presentavano la versione finale di Vidi 12, un digitalizzatore video dalle interessanti caratteristiche.

All'interno di altri stand non era difficile scorgere la sagoma di un Amiga, che veniva utilizzato per le mansioni più strane.

DCC contro MiniDisc

Quando si segue un mercato così attivo e proliferante come quello tecnologico, risulta a dir poco entusiasmante affrontare le problematiche dovute all'inserimento di nuovi prodotti: questa infatti è un'arena dove spesso solo la fortuna definisce il successo di uno standard su un altro. È accaduto spesso nella storia e l'esempio più citato in



La scheda Frame-Machine e Prism24 per i 16 milioni di colori

assoluto è il predominio del formato VHS su tutti gli altri pseudo-standard nel campo della videoregistrazione televisiva casalinga. Ciò che fa discutere più di ogni altra cosa, è il fatto che il VHS, sia come segnale video che come qualità audio, non aveva grandi vantaggi rispetto agli altri formati, Betamax o V2000 tanto per citarne alcuni... Ora questa appassionante guerra si sta per ripetere. Mi riferisco alla grande battaglia che si stanno dando la Philips e la Sony con i due nuovi formati per la registrazione audio digitale: il DCC e il MiniDisc.

Forse solo pochi si sono resi conto dell'importanza di queste due innovazioni tecnologiche. Dal punto di vista informatico, l'apporto rivoluzionario sarà decisamente grande: tenete infatti presente che entrambi i sistemi registrano in digitale. Ma quale dei due formati avrà la meglio? Al CeBIT '93 venivano presentati in diversi campi di applicazione, sia inseriti nel mondo musicale, sia in applicazioni computerizzate.

Sia la Sony che la Philips posseggono

case discografiche per cui si sono presentate sul mercato con diversi autori. Il DCC è un sistema a nastro magnetico dove i dati vengono letti e registrati in digitale (lo si può considerare una specie di DAT). Un punto di forza è la possibilità di leggere anche le vecchie audiocassette analogiche. Questo vuol dire che non bisogna necessariamente attendere l'uscita su DCC del disco del proprio cantante preferito, ma si possono utilizzare tutti i nastri già esistenti.

Il MiniDisc, invece, è la trasposizione della tecnologia floptical nel settore musicale. Il MiniDisc si presenta infatti scomposto su due facce, una magnetica (come i tradizionali floppy) e una ottica, simile a un tradizionale compact disc. I dati vengono registrati in digitale sulle tracce magnetiche, mentre la sezione ottica serve per allineare perfettamente la testina alle tracce. Un MiniDisc è in grado di contenere circa 75 MB di dati;

il costo di ogni disco dovrebbe aggirarsi attorno alle 30 mila lire, ma è destinato a scendere se questa nuova tecnologia avrà successo. Si assisterà quindi a una veloce rivoluzione nella tecnica di archiviazione dei dati: presto, sempre che lo standard diventi tale, i vecchi floppy verranno sostituiti dai nuovi MiniDisc.

Ad oggi mi sembra che nonostante le critiche sulla qualità audio dei MiniDisc, che è inferiore sia al DCC sia al normale CD, una grossa possibilità per questo supporto sia data dalle

potenzialità all'interno del settore informatico. Chi avrà la meglio solo il tempo può dirlo. È più probabile comunque che il DCC sfondi definitivamente nel campo dell'audio, mentre il MiniDisc sfrutti più ampiamente le sue caratteristiche digitali all'interno di applicazioni multimediali o comunque legate all'utilizzo dei computer.

Il Photo CD

Un'altra innovazione di cui forse pochi hanno capito l'importanza è il Photo CD della Kodak. Un lettore di Photo CD non è altro che un computer collegato a un lettore di CD-ROM studiato per visualizzare fotografie digitalizzate e registrate su CD su qualsiasi apparecchio televisivo; oltre alle caratteristiche video, è comunque anche un lettore CD audio a tutti gli effetti. Il problema è che la maggior parte di noi, limitandosi a guardare le fotografie su un televisore, nota soltanto gli aspetti negativi, quali la poca praticità (dover

cioè accendere diverse apparecchiature per vedere anche solo una fotografia), la freddezza delle immagini (per quanto dicano nella pubblicità le foto su carta sono molto più "calde", trasmettono più emozioni) e il costo del lettore. Così, però, ci si limita a vedere l'apparecchio in sé e non tutta la portata innovativa che ci sta dietro. Può essere invece interessante addentrarsi in quella che è la produzione del Photo CD.

Pur essendo relativamente recente, la tecnologia CD non aveva ancora assunto dimensioni commerciali tali da poter consentire a una larga fascia di persone di creare titoli, se non con l'ausilio di discrete disponibilità finanziarie. Mi riferisco alle varie applicazioni multimediali, ai programmi studiati per funzionare su piattaforme CD come il CDTV della Commodore e il CD-I della Philips. La tecnologia CD risultava molto conveniente in fase di duplicazione, ma dispendiosa nella creazione del disco master, detto anche Gold Disc. Pensiamo ora a quanto poteva costare fino poco tempo fa creare il Gold Disc, un disco ottico speciale sul quale viene inciso l'intero programma: ricordo che per un'applicazione del genere meno di due

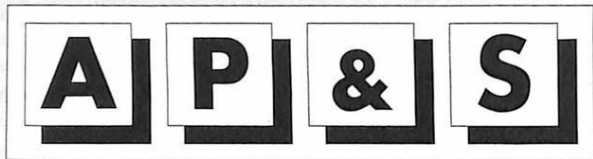
anni fa mi era stato chiesto qualcosa come due milioni di lire.

L'avvento del Photo CD e la produzione da parte della Philips di apparecchiature dedicate alla masterizzazione, caratterizzate dal costo contenuto, ha notevolmente abbassato i costi di produzione. A ottobre '92, allo Smau, era possibile creare un Gold Disc con sole 100 mila lire. Una riduzione decisamente entusiasmante, dovuta anche alla diffusione che queste apparecchiature stanno cominciando ad avere. Ma torniamo alla Kodak, che ha avuto la brillante idea di creare una rete di service dove, con la modica cifra di 20 mila lire, è possibile acquistare un Gold Disc e con 2 mila lire a foto è possibile farle incidere su disco. Immaginate ora di essere un esperto fotografo e di voler fare del fotoritocco, manipolare immagini fino a trasformarle in maniera entusiasmante; il primo passo da fare, dopo aver scelto le immagini, è quello d'inserirle in formato digitale all'interno del vostro computer. Questo passaggio significa disporre di apparecchiature costose e di una certa praticità con le stesse. Con il Photo CD la soluzione è invece presto trovata: scattate le vostre

foto, scegliete le più belle e trasferitele su CD. Vi sarete risparmiati tutta la difficoltà e l'onere delle apparecchiature, avendo comunque delle immagini di qualità decisamente superiore alla media. Pensate alle possibilità di utilizzo nel campo dell'impaginazione elettronica, nel desktop video... Quando poi sarà possibile creare anche Gold Disc di applicazioni multimediali a questi prezzi, la rivoluzione sarà ultimata.

Molti dei nostri lettori sono possessori di un CDTV, e una delle domande più inquietanti che si sono sempre posti è la possibilità o meno di utilizzare la tecnologia Photo CD sui propri apparecchi. La risposta non è ancora definitiva neanche da parte della Commodore, che deve decidere se acquistare o meno la licenza del software di gestione dei Photo CD dalla Kodak... Sembra che le ultime voci confermino che questa strada alla fine sarà imboccata: non resta che aspettare e vedere.

Avevo iniziato descrivendo le grandi novità e il successo di questa edizione del CeBIT. Sicuramente, ne sono tornato più che soddisfatto, convinto che il mercato Amiga stia crescendo sempre più nel campo professionale. ■



APPLIED PERIPHERALS & SOFTWARE

**DISTRIBUTORE PER L'ITALIA DEI PRODOTTI:
ACTIVA, DIGITAL CREATIONS, SUNRIZE INDUSTRIES**

SOFTWARE AMIGA

BRILLANCE	L. 390.000
MEDIALINK 3.0	L. 899.000
REAL 3D PROFESSIONAL 1.4.2	L. 199.000
REAL 3D 2.0	L. 999.000
REAL 3D 2.0 UPGRADE	TELEF.

HARDWARE AMIGA

AD 1012 + SOFTWARE STUDIO 16	L. 1.490.000
AD 516 + SOFTWARE STUDIO 16	L. 3.490.000
DCTV PAL	L. 790.000
DCTV PAL RGB CONVERTER	L. 490.000
MODULO SIMM 4Mb PER AMIGA 4000	TELEF.
ZIP RAM 4Mb PER AMIGA 3000	TELEF.

DISPONIBILI MANUALI IN ITALIANO

ATTENZIONE!! SCONTO DEL 5% PER PAGAMENTO ANTICIPATO

SI EFFETTUA LA VENDITA PER CORRISPONDENZA

CONSEGNA IN TUTTA ITALIA IN 24/48 ORE A MEZZO CORRIERE UPS (L. 29.000)

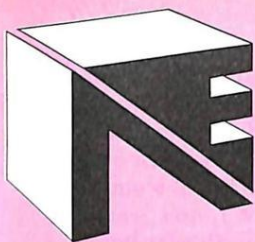
SI CERCANO PUNTI VENDITA

TUTTI I PREZZI SONO COMPRESIVI DI IVA E POSSONO SUBIRE VARIAZIONI CAUSATE DALL'ANDAMENTO VALUTARIO DEL DOLLARO E DEL MARCO.

Via Giovanni XXIII, 37
33040 Corno di Rosazzo
Udine - Italy
Tel. 0337-546686
Tel. 0432-759264
Fax 0432-759264

Orario di apertura:
Lunedì-Venerdì
09.00-12.00
13.30-18.30
Sabato 09.00-12.00

NON SOLOGRAFICA - UDINE

**NEWEL® srl****COMPUTERS ACCESSORI VIDEOGAMES**
20155 MILANO - VIA MAC MAHON 75**TEL. NEGOZIO (02) 39260744 (5 linee r.a.) FAX 24 ORE (02) 33000035 (2 linee r.a.)****ORDINA SUBITO**
02 - 33000036 (5 linee)**VENDITA
ANCHE PER
CORRISPONDENZA
IN TUTTA
ITALIA**

SCHEDE ACCELERATRICI MICROBOTICS

**VXL-30 25 MHZ A500/2000****L. 498.000****VXL-30 40 MHZ A500/2000****L. 548.000****VXL-32 MEMORY BOARD 2MB RAM BURST 32 BIT****L. 548.000****COPROCESSORE MATEMATICO PER VXL****L. 278.000**

Scheda acceleratrice per Amiga con coprocessore 68030 a 25 Mhz, zoccolo per coprocessore matematico 68882 con possibilità di espandere la memoria da 32 Bit fino a 8 Mb. Compatibile con il Burst del 68030 per ottenere 0 Wait State anche alle più alte frequenze di clock. Possibilità di commutare in modo 68000/68030 tramite software oppure tramite un interruttore. Non necessità di alcuna saldatura.

SCHEDE COPROCESSORI A1200 MBX MICROBOTICS

**MBX 1200 14 MHZ 881 CON OROLOGIO TAMPONE****L. 368.000****MBX 1200 25 MHZ 882 CON OROLOGIO TAMPONE****L. 528.000****MBX 1200 50 MHZ 882 CON OROLOGIO TAMPONE****L. 798.000**

Scheda coprocessore per gli Amiga 1200. Aumenta la velocità di calcolo del vostro Amiga. Indicato soprattutto per chi lavora con programmi tipo Cad 3D, Image ecc. che sfruttano tale coprocessore. La piastra è espandibile da 2 a 8 Mb tramite una porta SIMM a 32 Bit. Aumenta la velocità operativa di 50 volte se si aggiungono almeno 2 Mb di Ram a 32 bit. Non necessita di saldature, basta inserirla nello slot sotto l'Amiga. Corredata di Jumper per il Test e Software applicativo tra cui un test di velocità.

RAM 32 BIT DA 2,4 E 8 MBYTE PER A4000 E MBX 1200**MODULO SIMM DA 2 MB A 32 BIT****L. 288.000****MODULO SIMM DA 4 MB A 32 BIT****L. 488.000****MODULO SIMM DA 8 MB A 32 BIT (SOLO MBX 1200)****L. 899.000****12 A'CLOCK****L. 68.000**

L'Amiga 1200, lo sappiamo, è un computer perfetto eccetto per una cosa: necessità di orologio. Ora potrete avere un clock grazie alla scheda 12 A'Clock della Microbotics. Si installa nello zoccolo interno dell'Amiga senza saldature. Potrete così utilizzare tutte quelle utility che si servono dell'orologio come: la sveglia del vostro Workbench, datare i file, programmi con autosaving a tempo ecc.

ATTENZIONE! NOVITÀ KIT HARD-DISK AMIGA 1200

(IL KIT COMPRENDE HD + SOFTWARE + CAVO)**20 MB**
30 MB
40 MB**L. 280.000**
L. 340.000
L. 390.000**820 MB**
120 MB
180 MB**L. 690.000**
L. 980.000
L. 1.350.000

TUTTI I NOMI E I MARCHI SOPRA RIPORTATI APPARTENGONO AI LORO PROPRIETARI. I PREZZI SONO TUTTI IVA COMPRESA. SUPERGARANZIA PER 1 ANNO.
TUTTI I PRODOTTI MICROBOTICS SONO DISTRIBUITI IN ITALIA DA NEWEL srl.

PROVE HARDWARE

CLARITY 16: LE NUOVE FRONTIERE AUDIO DELL'AMIGA

Questo nuovo prodotto per Amiga offre una digitalizzazione audio a livelli qualitativi mai raggiunti finora dagli apparecchi di questa fascia di prezzo

di Lorenzo Fornari

È finalmente disponibile sul mercato, importato dalla ComputerLand di Milano, l'atteso campionatore audio a 16 bit Clarity 16 della ditta inglese Microdeal. Questa volta l'impressione è che il mondo Amiga possa sfidare anche i più agguerriti concorrenti nel campo della computermusica, forse finora un po' trascurato, e a un prezzo fino a poco tempo fa impensabile per un certo tipo di prestazioni. Che si fosse prossimi a una svolta decisiva lo si era capito già da tempo per le sempre più insistenti voci di un imminente rivoluzionario dei circuiti audio dell'Amiga: e anche se la cosa è stata momentaneamente rinviata, il crescente interesse degli utenti Amiga per le applicazioni musicali lasciano presagire grandi sviluppi in questo campo. Non poteva dunque esserci un momento più favorevole per l'apparizione di Clarity 16. I risultati ottenibili con questo campionatore, infatti, sono al di là dello standard a cui l'utente Amiga è abituato. Tenete solo presente che il sistema di campionamento dei

compact disc è anch'esso a 16 bit, e che mentre questi hanno la frequenza di campionamento fissata a 44.1 KHz, il campionatore della Microdeal vi mette in condizione (con Amiga veloci, quindi non con A500, A600 o A2000 non accelerati) di campionare fino a ben

precisazione non inutile: quando si parla di campionamenti a questi livelli qualitativi spesso si trascurava il fatto che la qualità di un campionamento dipende dal segnale in entrata: infatti un conto è campionare da un compact disc DDD letto da un impianto costato una decina

di milioni, mentre un'altra cosa è campionare la propria voce tramite un microfono collegato al proprio stereo... Per questa prova è stato impiegato un Amiga 1200 e svariate fonti, principalmente un lettore di compact disc portatile Technics e un microfono Shure preamplificato da un mixer Fostex; ulteriori verifiche sono state effettuate con un'attrezzatura più

professionale, costituita da un banco a 16 canali H&R, microfoni Electro-Voice, lettore di CD Sony, tastiera Roland D-50 e amplificatore Pioneer da 750 watt con quattro casse JBL. Il risultato è stato di una fedeltà pressoché totale: assenza di rumore di fondo e chiarezza cristallina del suono; riproduzione, insomma, fedele nei minimi particolari.

Naturalmente, però, le massime pre-



59.9 KHz! Proprio per garantire le massime prestazioni, quindi, il prodotto è stato fornito di uscite audio autonome ad alta fedeltà, grazie alle quali è possibile apprezzarne appieno le potenzialità (la circuiteria di riproduzione audio dell'Amiga è attualmente a 8 bit, mentre quella Clarity 16 è a 16 bit).

Ma prima di descrivere dettagliatamente il campionatore, bisogna fare una

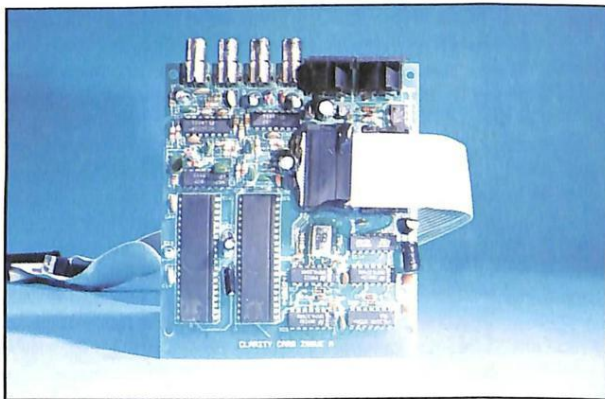
stazioni si ottengono con i nuovi Amiga o con quelli accelerati. Con un A500, A600 o A2000 si riesce a campionare fino a un massimo di circa 44 KHz in mono e 32 KHz in stereo. Nelle nostre prove con l'A1200 siamo riusciti a impiegare frequenze fino a circa 42 KHz in stereo e oltre i 50 KHz in mono. Come dice anche il manuale, con gli Amiga veloci come l'A1200 è però necessario attivare la funzione "fast Amiga", che richiede l'inserimento di tre parametri numerici di cui uno a tre cifre, i quali però hanno dei valori di default errati che ci hanno causato problemi di ogni tipo: blocchi del sistema, campionamento con fortissimi rumori di fondo, meter inutilizzabili... Dopo moltissimi tentativi abbiamo poi trovato una terna magica (1, 78, 66) con la quale Clarity 16 ha iniziato a funzionare decentemente e abbiamo potuto proseguire con la prova che state leggendo.

Il package e l'hardware

Il prodotto, garantito un anno dal costruttore, si presenta come un'interfaccia esterna di dimensioni ridotte (13 x 11 x 4 cm) color vaniglia da connettere all'Amiga tramite le porte parallela e seriale contemporaneamente, per garantire la massima precisione e velocità nel passaggio delle informazioni. Non spaventi questa invadenza di Clarity 16, perché, benché occupi tutte le porte disponibili, questa interfaccia contiene tutto quello di cui il musicista Amiga può aver bisogno. Sul retro della scheda, infatti, si trovano quattro prese RCA (due AUDIO IN e due AUDIO OUT), come si è già detto ad alta fedeltà. Ma non è tutto, perché Clarity 16 è fornito di due prese pentapolari MIDI (IN e OUT, ovviamente) che, oltre a espandere notevolmente le possibilità del campionatore, sono compatibili con la maggior parte del software musicale per Amiga. E a chi obiettasse l'assenza di una porta THRU mi sento di ricordare che la maggior parte dei programmi di sequencing permette di ovviare a questo problema tramite la funzione chiamata "midi merge", che trasmette attraverso la porta MIDI OUT tutte le informazioni ricevute attraverso il MIDI IN, permettendo

quindi esattamente la funzione di solito adempiuta dal MIDI THRU.

L'interfaccia si collega al computer tramite due "flat cable" che in virtù della loro lunghezza (circa mezzo metro), consentono di alloggiare l'unità nella posizione voluta. Ho anche aperto il box di Clarity 16 per verificare la precisione dei circuiti, e ho notato che lo stampato è sottilissimo; e se da una parte questo riduce pericolosamente la robustezza, dall'altra rappresenta un'ottima misura difensiva da qualsiasi interferenza: si tratta quindi di un'altra conferma del fatto che ci troviamo di fronte a un



Sopra: in primo piano le porte MIDI, gli ingressi audio e le uscite audio a 16 bit. Sotto: la circuiteria del Clarity 16

apparecchio ad alte prestazioni, e che, come tutti i moduli ad alta fedeltà, Clarity 16 è molto delicato. In particolare, oltre alle ovvie cautele, il manuale suggerisce di prestare molta attenzione quando si estraggono i pin dalle prese, evitando di scuoterli per evitare di estrarre con il cavo anche parte dell'interfaccia. Si tratta comunque di un caso limite, perché Clarity 16 mi pare in definitiva piuttosto robusto, mentre il consiglio del manuale dovrebbe essere seguito non solo quando si maneggia questo campionatore. È anzi una buona norma da applicare ogni qualvolta si abbia a che fare con apparecchiature elettroniche in genere.

Il manuale e il software

Il manuale, che è in inglese, è troppo breve e non del tutto esauriente. Spesso la brevità va a scapito della chiarezza, anche se fortunatamente l'uso del programma è facilitato dal fatto che ormai determinate funzioni dei campionatori sono diventate standard, sia a livello di nome che di modalità d'uso, cosicché chi abbia anche una minima familiarità con l'uso di un altro campionatore può venirne a capo in tempi veramente rapidi; e si tratta, fortunatamente per i neofiti della digitalizzazione sonora, di una o due eccezioni. Resta comunque il fatto che alcuni potrebbero essere infastiditi da frasi del tipo "si potrebbero dare ulteriori spiegazioni, ma la cosa migliore la vostra sperimentazione diretta", quando invece qualche riga di chiarimento in più non avrebbe fatto male a nessuno.

Il software è rappresentato da due dischi, il primo contenente il programma, l'altro i file demo. A proposito di questi ultimi, mi sono molto meravigliato della loro esiguità e del fatto che non siano per nulla impressionanti: e dal momento che l'acquirente non ha altra possibilità di provare il programma se non ascoltando i file demo, allora mi sembra di poter dire che siamo di fronte a una scelta commerciale abbastanza miope.

Caratteristiche e requisiti

La compagine più esigente degli utenti Amiga si meraviglierà del fatto che né sul manuale né altrove sono presenti le caratteristiche tecniche specifiche di Clarity 16. Insisto su questi aspetti, anche se magari viste le potenzialità del prodotto ci vorrebbe molto poco per renderlo molto migliore. Inoltre, ho parlato di un prezzo molto basso, ma si tratta sempre di 350 mila lire, per cui mi pare lecito esigere una certa qualità anche nelle finiture e nei particolari meno importanti.

Importantissimo è invece il fatto che, come abbiamo già detto, se possedete un computer che rientra nella categoria definita dal manuale "fast Amigas" (A1200, A3000...) il programma tende

a bloccarsi facendo uso di determinate funzioni. Qui la lacuna del manuale è veramente grave, perché, spiegando che nel caso si verificano ripetuti blocchi del sistema bisogna andare a modificare tre parametri di facile accesso, commette la "leggerezza" di non spiegare né cosa indichino esattamente questi parametri, né come essi vadano variati! La

cosa migliore da fare è chiedere garanzie al vostro negoziante o contattare l'importatore prima dell'acquisto per essere sicuri della compatibilità del proprio Amiga con il programma di Clarity 16. Mi sento di dover dare questo consiglio perché le riviste in lingua inglese affermano, per esempio, che Clarity 16 non è adatto a lavorare alla

velocità dell'Amiga 4000. E in questo caso allora ci chiediamo: ma con quali Amiga si possono raggiungere i tanto decantati 59.9 KHz, visto che per esempio con l'A1200 non si superano i 42/44 KHz in stereo?

Clarity 16 ha bisogno di almeno 1 MB di memoria, ma ovviamente più se ne ha meglio è. Con l'A1200 non ho notato

LA DIGITALIZZAZIONE AUDIO

Fino a qualche anno fa l'unico modo di registrare una fonte sonora era quello analogico, basato sulla trasformazione del suono in una serie di variazioni di campo elettrico, poi registrate mediante apparecchiature magnetiche. Il risultato viene rappresentato sotto forma di onda sinusoidale: esattamente come il suono "vero", con il tempo sull'asse delle ascisse e l'intensità sull'asse delle ordinate; la differenza è che mentre nel suono reale questa intensità si riferisce alla pressione dell'aria sulle nostre orecchie, nel suono analogico essa è riferita all'intensità del campo elettrico. Questo sistema al giorno d'oggi ha raggiunto una notevolissima evoluzione, tuttavia presenta tutta una serie d'inconvenienti legati al fatto che quando si ha a che fare con campi elettrici si è sempre soggetti a interferenze, che deteriorano il segnale in fase di registrazione e di ascolto.

Per ovviare a questi problemi, si è evoluto abbastanza rapidamente un sistema alternativo di registrazione, chiamato appunto digitale. Esso, anziché basarsi su variazioni di campo elettrico, si basa sulla descrizione della forma d'onda mediante una sequenza di numeri che viene poi registrata con il codice binario. I vantaggi sono rappresentati dal fatto che questa serie numerica sarà sempre uguale a se stessa, senza alcun tipo d'interferenza elettromagnetica. Inoltre, è possibile modificare questa sequenza tramite algoritmi matematici, senza arrecare la benché minima attenuazione al segnale originale; in pratica, la fase di edit del suono digitalizzato non presenta quegli inconvenienti di progressivo deterioramento tipiche dell'edit analogico.

Vediamo ora a grandi linee come funziona la digitalizzazione sonora. Il suono di partenza è sempre analogico, per esempio proveniente da un microfono. Esso viene convertito in digitale secondo due parametri: la frequenza di campionamento e la risoluzione di campionamento. La frequenza di campionamento si misura in Hertz e indica quante volte il suono viene campionato nello spazio di un secondo. La peculiarità del sistema di registrazione digitale, infatti, è che la registrazione non è continua (come nella registrazione analogica), bensì istantanea; questo significa che quello che generalmente chiamiamo campionamento in realtà è la giustapposizione di moltissimi campionamenti istantanei. L'esempio che solitamente si porta è quello della pellicola cinematografica: anch'essa è infatti una successione di eventi istantanei (i singoli fotogrammi), che però a determinate condizioni (cioè se i fotogrammi sono molto frequenti, in parole povere se la velocità della cinepresa è abbastanza alta) ci danno un'impressione di continuità. Ora, quello che succede quando noi campioniamo un suono in maniera digitale è simile a quello che fa una normale cinepresa: mentre questa a intervalli regolari scatta una fotografia "visiva", il campionatore scatta a intervalli regolari delle "fotografie sonore". E come poi la cinepresa ci dà un'impressione di continuità mostrandoci in velocissima sequenza le singole immagini, non diversamente si comporterà coi suoni il nostro digitalizzatore; infine, come il risultato visivo sarà più o meno realistico a seconda della maggiore o minore velocità della cinepresa (cioè quanti fotogrammi al minuto riesce a scattare), così avverrà per i suoni, e questa "velocità" è appunto la frequenza di campionamento. Va da sé che più essa è alta, più realistica sarà l'immagine sonora che otterremo. Qualcuno potrebbe allora chiedersi perché i compact disc, che paiono essere uno dei più sorprendenti risultati in questo campo, abbiano la frequenza di campionamento limitata a 44.100 Hz. Il fatto è che l'orecchio umano non può percepire frequenze al di là di una determinata soglia, mediamente circa 20.000 Hz (meglio una precisazione subito per

evitare fraintendimenti: quella che ho chiamato frequenza di campionamento rappresenta il numero di campionamenti effettuati in un secondo; la frequenza di cui parlo adesso è quella che si riferisce all'altezza di un suono, cioè al fatto che esso sia più o meno acuto): questa limitazione rende del tutto inutile la registrazione di segnali di frequenza superiore a questa soglia. Ora, tutto questo era per giustificare la frequenza di campionamento (di campionamento, nota bene) dei compact disc, limitata 44.100 Hz. Il fatto è che è stato formulato, e dimostrato, un teorema in base al quale quando campioniamo un suono di una certa frequenza (cioè altezza), è sufficiente che la frequenza di campionamento del digitalizzatore sia doppia dell'altezza del suono. Questo significa che se noi dobbiamo campionare un suono dell'altezza di 256 Hz ci basterà campionare a una frequenza di 512 Hz; tuttavia bisogna stare attenti a non generalizzare troppo, perché quella che noi chiamiamo frequenza di un suono è soltanto la frequenza principale del suono in questione, che poi ha delle cosiddette armoniche (che così su due piedi potremmo definire, continuando la metafora visiva di prima, le "sfumature" del suono) di frequenza superiore a quella principale. In ogni caso, è certo che non siamo in grado di udire suoni, o armoniche che siano, di frequenza superiore a 20.000 Hz, per cui è perfettamente inutile, sempre secondo questo teorema, campionare a oltre il doppio di questa frequenza.

Per quanto riguarda quella che ho chiamato risoluzione di campionamento, normalmente standardizzata a 16 bit (nei compact disc) o anche a 24 in alcune apparecchiature professionali, essa ha a che vedere con la precisione con cui ogni singolo campionamento viene registrato. Campionare a 8 bit significa avere una rosa di 256 eventi possibili riguardanti l'intensità, che è l'altra componente fisica del suono oltre all'altezza; campionare a 16 bit significa espandere questa rosa a ben 65536 eventi. Tutto questo ha a che fare con la cosiddetta quantizzazione: non tutte le variazioni d'intensità vengono registrate così come sono realmente, ma solo quelle che coincidono con uno dei livelli di risoluzione; gli altri vengono modificati in modo da coincidere con il più vicino di questi livelli. Cercherò di spiegarvi un po' meglio con un esempio: campionare a 8 bit è come disporre un oggetto sopra una scalinata alta 100 metri e composta da 256 scalini. Ammettiamo che l'oggetto debba essere posto a un'altezza di 50 metri dal suolo: esso ovviamente andrà posto sul centoventinovesimo scalino. Ma se invece l'oggetto dovesse essere posto a un'altezza di 49,999 metri, come ci comporteremmo? Dovremmo operare una "quantizzazione", cioè arrotondare il valore richiesto fino a farlo coincidere con un valore corrispondente a uno scalino. Questo è più o meno quello che accade nella risoluzione digitale: tra gli infiniti valori possibili d'intensità il digitalizzatore ne sceglie soltanto alcuni (256 se si campiona 8 bit), disposti a intervalli regolari, e poi arrotonda i valori d'intensità non esattamente corrispondenti a questi "scalini" a uno dei valori di questi ultimi. Nel caso del campionamento a 16 bit l'esempio sarebbe quello di una scala alta sempre 100 metri, ma dotata di 65536 scalini: avremmo certamente molto meno bisogno di arrotondare, e il posizionamento del nostro oggetto sarebbe molto più accurato; così viene da sé che più alta è la risoluzione, più fedele è il risultato.

Per concludere, quindi, la frequenza di campionamento indica il numero di volte all'interno di un secondo che il digitalizzatore registra un'"immagine" della sorgente sonora, mentre la risoluzione di campionamento è la capacità di riprodurre più o meno fedelmente ciascuna di queste immagini.

alcuna anomalia di funzionamento, e credo di averlo provato con il computer più adatto alle caratteristiche del campionatore. A una prima analisi, mi era parso che avere l'hard disk fosse perfettamente inutile, perché il programma sta tutto in un comodo dischetto, e d'altra parte l'archiviazione dei campionamenti è più comoda su floppy. A queste condizioni, però, bisogna stare attenti a non fare campionamenti troppo lunghi, visto che la capacità di un dischetto è di 880K; e quando si campiona a 16 bit in stereo è veramente facile eccedere questo apparentemente ampio margine.

Il campionamento

Avviando il programma, la prima cosa che salta all'occhio è l'assenza di una schermata propria, per cui l'apertura del programma avviene nella finestra di lavoro del *Workbench*: questo permette di risparmiare preziosi byte, visto che per quanto possiate disporre di una vasta memoria, i campionamenti stereo a 16 bit hanno bisogno di più spazio possibile; è anche vero però che non vi è la possibilità di poterne creare una nel caso si fosse disturbati dall'apparente disordine che viene a crearsi sullo schermo. In ogni caso, se risultasse indispensabile il funzionamento su finestra propria, è comunque possibile utilizzare utility di pubblico dominio, come *PSX* di Steve Tibbett.

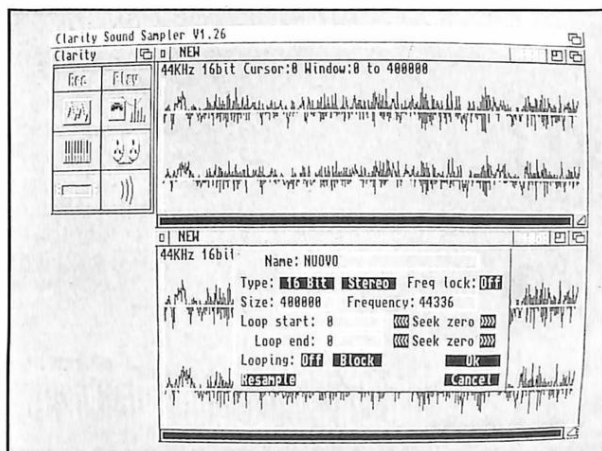
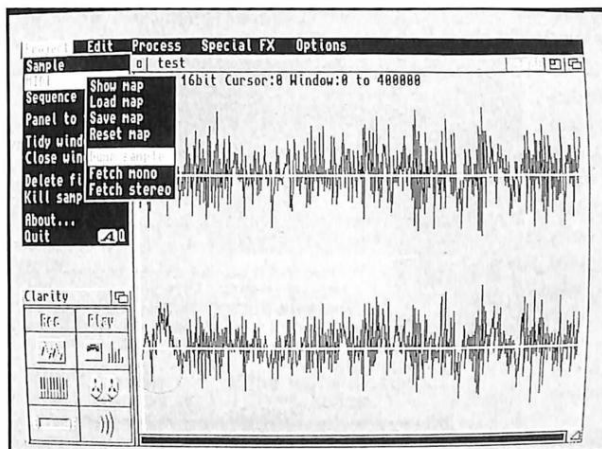
Caratteristica principale della grafica di Clarity 16 è l'organizzazione dello spazio video in molte piccole finestre, ognuna dedicata a una funzione particolare. La prima schermata contiene una di queste finestre, che dà accesso alle funzioni principali: in primo luogo ai *METER* audio. Potete scegliere tra mono e stereo, e tra tre tipi diversi di meter: da quelli *VU* (a lancette, per intenderci), a quelli oscilloscopici, a quelli a rappresentazione grafica di banda. Questi meter sono molto utili quando si prova il volume d'ingresso per evitare inutili distorsioni oppure l'opposto difetto di campionamenti a volume troppo basso, ma purtroppo non permettono di controllare i volumi durante il campionamento vero e proprio.

Il riquadro iniziale permette di accedere anche ad altre opzioni, quali la

visualizzazione dei campionamenti in memoria, un'utile icona con la quantità di memoria rimasta libera, un basilare sequencer e un set di effetti applicabili ai campionamenti o anche al suono in entrata, in tempo reale: tutte queste funzioni sono accessibili tramite apposite finestre che si dispongono in modo vario sullo schermo; grandezza e posizione sono a scelta dell'utente tramite il comando "program config". A proposito degli effetti speciali, sono disponibili:

applicarle e successivamente valutarle, e l'operazione non è reversibile (niente "undo", in parole povere); la sola maniera di procedere è quindi quella di effettuare un backup della forma d'onda prima d'introdurre questa funzione. Ma poiché queste funzioni sono posteriori alla registrazione di un campionamento, sarà meglio occuparsi prima della fase di campionamento vero e proprio.

L'operazione di campionamento è di estrema semplicità: basandovi sui meter d'ingresso regolate il volume della vostra fonte sonora; aprite il menu "project" e selezionate "sample new". A questo punto vi trovate di fronte al preset di campionamento scelto dalla Microdeal, che include nome del campionamento, frequenza e risoluzione di campionamento, durata della registrazione e varie altre opzioni relative al loop. Per prima cosa, abbiamo provato un campionamento a 16 bit in stereo, a una media frequenza, circa 30 KHz, per poter poi disporre della maggior memoria possibile in fase di edit. Usciti dalla finestra di preset, abbiamo fatto partire la registrazione con il mouse; si può anche impiegare l'opzione trigger che, se inserita, fa partire la registrazione solo dopo che vi sia un segnale in entrata con un volume superiore a una certa soglia (threshold). Prima di riascoltare, è buona norma utilizzare la funzione "normalise", che porta il volume a livelli ottimali nel caso il campionamento sia venuto un po' debole. È indubbio: il risultato è sorprendente.



Sopra: quella evidenziata è la funzione che consente di trasferire i campionamenti a 16 bit tramite le porte MIDI. Sotto: l'interfaccia principale con due campionamenti attivati

La fase di edit

riverbero, flanger, chorus e distorsore. A parte l'eco (che è ovviamente impressionante, essendo a 16 bit), gli altri effetti hanno un impiego molto limitato se applicati direttamente al segnale in entrata (il distorsore per esempio è stato provato con una chitarra elettrica con dubbi risultati), mentre offrono ampie possibilità quando si tratta di modificare dei campionamenti o parte di essi. In due parole, si tratta di effetti che lasciano a desiderare come real time processor, mentre soddisfano pienamente l'utente in fase di edit. Purtroppo, queste operazioni non si possono provare per poi decidere se attivarle: si può solo

A questo punto si entra nella fase di edit: per apprezzare appieno le potenzialità di Clarity 16 è essenziale familiarizzare con la "sample window"; e forse qui sta la sorpresa più piacevole: il massimo della chiarezza e semplicità d'uso. Infatti, dopo poche prove si entra immediatamente nell'ordine d'idee necessario ad apportare modifiche anche sostanziali a qualsiasi tipo di campionamento, che viene presentato graficamente sullo schermo in maniera essenziale e al quale sono applicabili facili comandi che rendono accessibile l'edit della forma d'onda anche al più sprovveduto degli utenti.

La gestione della forma d'onda avviene tramite la selezione di una parte del campionamento (o anche di tutto) con il mouse sulla finestra del campionamento: prende il nome di "block", e su di esso potete operare tutte le classiche modifiche dei campionatori, a cominciare dall'aggiungere uno degli effetti a disposizione, oppure invertendo la forma d'onda, o ancora prevedendo delle ripetizioni del singolo "block" (il cosiddetto "loop"), mischiare tra loro due forme d'onda diverse (ossia due block) e moltissime altre opzioni.

Innanzitutto, va notata la possibilità di operare un progressivo zoom fino ai minimi dettagli, che permette un edit minuzioso, soprattutto se unita alla funzione rappresentata dal comando "DB" (che aumenta graficamente il volume del campione per poter meglio visualizzare i particolari più piccoli). In ogni momento è possibile ascoltare la sezione selezionata con il mouse e chiaramente evidenziata sullo schermo, oppure far partire il playback da un determinato punto, per rendervi conto se state operando sulla parte di campionamento desiderata. Per la creazione dei loop, c'è addirittura la funzione "Seek zero" che va alla ricerca dei "punti morti" del campionamento, facilitando notevolmente il lavoro. L'edit vero e proprio, poi, avviene attraverso il menu "Process", che tra le altre cose permette d'iniziare o terminare un campionamento o una sua sezione con un cosiddetto "fade" (cioè un graduale innalzamento del volume di tutto il block), di scambiare i canali fra loro e molte altre operazioni.

Inoltre, c'è la possibilità di vedere in una finestra tridimensionale la forma d'onda del vostro campionamento tramite il display FFT, oppure optare per una millimetrica "Freehand edit" su una rappresentazione grafica adatta allo scopo, oppure di scegliere, ed eventualmente modificare, i filtri a disposizione. Questi ultimi non si limitano solo al filtro passa-banda di cui ho parlato all'inizio, ma sono quattro: low-pass, high-pass, band-pass e band-stop. Da ultimo, è consigliabile dividere i campionamenti in "section" (fino a dieci), tramite un semplice comando: queste sezioni sono salvate infatti insieme al campionamento, e potranno rivelarsi utili in seguito.

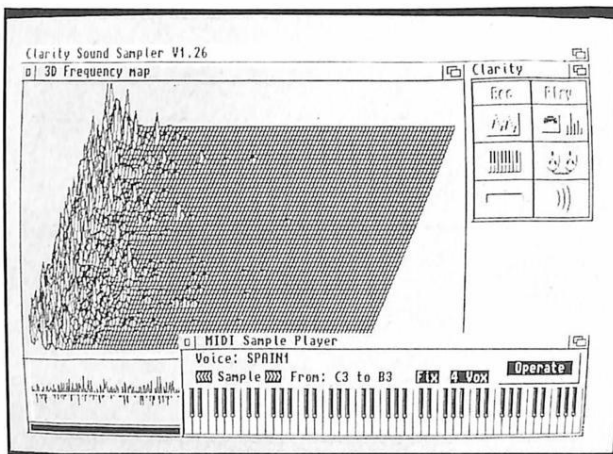
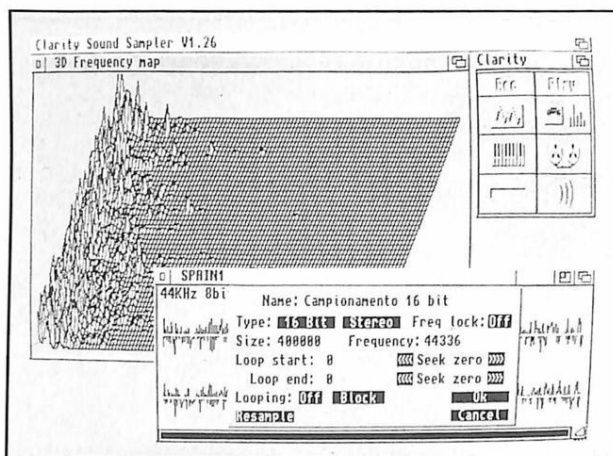
Una volta scelti ed editati i campionamenti, si può cominciare a divertirsi con il sequencer incorporato. A questo proposito, non bisogna farsi sviare dall'uso che generalmente si fa della parola "sequencer": non si tratta infatti di niente di paragonabile a KCS o a Bars & Pipes, ma soltanto di un "sequenziatore". Infatti, esso è in grado di far suonare esclusivamente un campionamento per volta, o parte di esso, per poi passare a un altro, secondo una sequen-

map" (ovviamente salvabile su disco); suonando poi i tasti della tastiera, si fanno suonare i campionamenti, variandone l'accordatura a piacimento, e con polifonia da 1 a 4 campionamenti contemporanei. Logicamente, si ottimizza questa funzione utilizzando un sequencer esterno a Clarity 16; l'uso della tastiera esterna non è comunque inevitabile, visto che si possono suonare i campionamenti a piacimento tramite i tasti funzione dell'Amiga e variare l'accordatura tramite la tastierina

numerica. Questa funzione, oltre a essere molto utile durante il playback (Clarity 16 può tenere fino a 64 campionamenti in memoria, RAM permettendo!), risulta utile anche da un altro punto di vista: il campionatore include un'interfaccia MIDI adatta a funzionare con moltissimi programmi musicali Amiga. Ma l'uso più particolare per cui questa interfaccia MIDI è stata studiata è lo scambio d'informazioni con un campionatore esterno, come per esempio un Akai S1000: Clarity 16 carica da esso un campionamento, dopo di che lo si può editare graficamente con tutta la comodità d'uso caratteristica di Clarity 16, per poi infine richiamare con l'S1000 il campionamento così modificato. Naturalmente, Clarity 16 può scambiare i propri campionamenti non solo con campionatori esterni all'Amiga, ma può anche far suonare i propri campionamenti a programmi di sequencing che accettino i suoni in standard IFF; infatti, Clarity 16 ha tre modi possibili di "sample saving": IFF (lo standard a 8 bit usato finora dall'Amiga), AIFF (cioè un IFF a 16 bit, forse lo standard della

prossima generazione Amiga) e AVR (uno standard che permette di salvare anche le informazioni relative al campionamento, le "section", e altre informazioni che vengono trascurate dai due standard precedenti). Il consiglio è di salvare sempre in AVR, che poi può eventualmente essere trasformato in uno dei due precedenti formati, mentre non è possibile fare l'inverso (cioè formattare un IFF in un AVR).

La cosa che più colpisce l'udito è però l'estrema qualità del suono, anche usando le uscite audio Amiga (nel caso si usi un sequencer interno al proprio calcolatore, oppure monitorizzando il proprio



Sopra: un campionamento visualizzato in Fast Fourier (si noti la finestra di campionamento). Sotto: la fase di assegnamento dei campionamenti ai singoli tasti della tastiera MIDI

za impostata dall'utente, con anche la possibilità di pause, ripetizioni o playback di una "section" soltanto. È ovviamente possibile salvare la propria sequenza; è inoltre disponibile una sequenza dimostrativa, per la verità non molto riuscita, che rende chiaro un possibile uso di questa applicazione per esempio per la sonorizzazione di propri filmati video con effetti audio.

Un'altra funzione extra è la cosiddetta "MIDI", che permette di assegnare a una tastiera collegata via MIDI i campionamenti di Clarity 16. Tramite semplici operazioni, si può assegnare a ogni tasto un campionamento, creando una "midi-

UNO SGUARDO ALLA CONCORRENZA

Mi sembra inevitabile dare una rapida occhiata al mercato dei campionatori per Amiga, rimandando ai numeri precedenti di questa rivista per chi volesse approfondire l'argomento: molti di essi sono infatti stati recensiti (si vedano in particolare i numeri 1/91 e 6/91 di *Commodore Gazette*).

In primo luogo, mi sembra doveroso cominciare con quello che da tutti è riconosciuto come il miglior campionatore a 8 bit per Amiga, cioè Audio Engineer Plus della Ramscan. Nonostante il suo prezzo decisamente alto se confrontato con prodotti dalle caratteristiche simili, questo prodotto per precisione dell'hardware e sofisticatezza del software può definirsi a tutti gli effetti un campionatore di livello quasi professionale. Sempre restando nel campo dell'alta qualità a 8 bit, non si può poi dimenticare l'altrettanto famoso *AudioMaster IV* della Aegis: si tratta, attualmente, del più acclamato software per la gestione di campionamenti tra quelli esistenti, e non a torto, molte case di produzione hardware, infatti, progettano le loro interfacce proprio pensando ad *AudioMaster* come software ideale. Anche qui però, il prezzo è piuttosto alto per offrire soltanto del software, per quanto pure in questo caso ci troviamo di fronte a un programma di uso quasi professionale. Le leggi della concorrenza, poi, non potevano certo permettere che un prodotto fortunato come *AudioMaster* non fosse subito oggetto d'imitazione per tentare di scalzarlo dal suo ruolo guida nel campo della gestione delle forme d'onda digitali: sono nati così moltissimi programmi dello stesso tipo. Uno molto riuscito, che non può certo essere considerato semplicemente un epigono di *AudioMaster* (anche per le interessanti innovazioni del tutto autonome rispetto al programma della Aegis) è *Audition 4* della Sunrize. Un punto a favore di *Audition* è il prezzo, un po' più contenuto di quello di *AudioMaster*.

Per rimanere su apparecchiature di alta qualità, a questo punto è necessario parlare di due prodotti della Sunrize, avvertendo però che siamo di fronte a due autentici fuoriserie, sia per quel che riguarda la qualità, ma soprattutto per quanto concerne il prezzo di vendita. Il fatto è che questi finora erano gli unici campionatori per Amiga ad aver sfondato la barriera degli 8 bit: fatto che se da una parte, come abbiamo visto, garantisce risultati al di là di ogni aspettativa, ha però suggerito ai distributori di fissare un prezzo di ben oltre un milione di lire! Il primo, l'AD1012 (1.490.000 lire presso la AP&S, Tel. 0432/759264) è un campionatore a 12 bit che tra l'altro garantisce anche la possibilità di registrare sull'hard disk fino a quattro tracce separate per poi risentirle contemporaneamente: un vero e proprio registratore multitraccia, quindi. Il secondo prodotto (3.490.000 lire), si chiama AD516 ed è la versione a 16 bit dell'AD1012. Il numero di tracce gestite è qui però di otto. Entrambe le schede lavorano con un potente software che prende il nome di *Studio 16*.

Non c'è musicista Amiga invece che non abbia provato almeno una volta il famosissimo SoundMaster della Aegis: sia per la contenutezza del prezzo (anche se qui siamo di nuovo a 8 bit), sia per l'affidabilità generale del prodotto, pensato per lavorare con *AudioMaster*.

Infine, devo accennare alla vera e propria proliferazione di mini-campionatori di prezzo contenutissimo, tutti ad 8 bit e di pretese non eccessive, ma perfetti per il principiante che con una spesa quasi irrisoria può cominciare a familiarizzare con il campionamento, per poi però passare ad apparecchi più curati quando voglia fare qualcosa di più serio. Concludo con un prodotto che per una curiosa caratteristica merita un posto di rilievo tra i campionatori a 8 bit meno conosciuti: si tratta di Sampler Plus, hardware progettato per lavorare con *AudioMaster*. Questo campionatore, che peraltro quanto a qualità sorpassa di gran lunga i cosiddetti "minori", in fase di riascolto utilizza le uscite audio dell'Amiga, come molti dei campionatori di cui ho parlato; tuttavia è dotato di un preamplificatore che permette di ascoltare i suoni in cuffia collegandosi direttamente all'interfaccia; e questo non solo per quel che riguarda i campionamenti, ma anche qualsiasi suono sia prodotto dall'Amiga durante l'utilizzo di qualunque programma: una caratteristica interessante, per chi sia appassionato di videogiochi, vista l'evoluzione della controparte musicale di questi ultimi. Ma non è tutto, perché Sampler Plus è un prodotto tutto italiano, essendo fabbricato dalla PG Elettronica di Torino.

sample in fase di registrazione, o anche solo per curiosità si possono e in alcuni casi si devono usare le uscite audio del calcolatore). Probabilmente, questa eccezionale qualità è dovuta al fatto che il software di Clarity 16 utilizza le uscite audio dell'Amiga (notoriamente a 8 bit) accoppiate, in modo da ridurre le capacità polifoniche, aumentandone però notevolmente le potenzialità qualitative.

Conclusioni

Clarity 16 è troppo "prepotente" in

fase di multitasking, vanificando così la possibilità di registrare direttamente su hard disk, utilizzando la cosiddetta "memoria virtuale". Il difetto al momento più grave (che è ragionevole ritenere venga risolto al più presto, vista la disponibilità dei nuovi Amiga) riguarda la questione dell'adattamento del programma alla velocità del processore dei nuovi Amiga o degli Amiga accelerati, che a nostro avviso dovrebbe essere del tutto automatica e invisibile all'utente: un'apposita finestra dovrebbe avvisare riguardo alle frequenze massime impiegabili sulla base della frequenza di clock

disponibile.

Concludo però con una nota positiva: la versatilità di Clarity 16, grazie alla quale è possibile impostare a proprio piacimento tutti i parametri che si vuole (configurazione MIDI, fast-Amiga parameters, addirittura i colori usati dal programma), con la possibilità di salvare il tutto. Inoltre, il fatto di aver raggiunto notevoli risultati con soli 2 MB di RAM e senza hard disk, mostra che nonostante le grandi prestazioni possibili, il campionatore è destinato a una vastissima fetta di pubblico, anche in virtù del suo prezzo. Se infatti l'alta qualità dei campionamenti può soddisfare il più esigente degli utenti che ne faccia uso anche semi-professionale, tuttavia la semplicità d'uso viene incontro a chi per la prima volta si avvicina al mondo della digitalizzazione sonora. E del resto il prezzo abbordabile con l'aggiunta dell'interfaccia MIDI rende Clarity 16 un prodotto veramente interessante.

**Per ulteriori informazioni
contattare direttamente:**

ComputerLand
(Clarity 16: L. 350.000 Iva compresa)
Via Cenisio 55/C
20154 Milano
(Tel. 1fax 02/33104236)

SCHEDA CRITICA

Prodotto:

CLARITY 16

VOTO: **7,0**
(In decimi)

Funzionalità:	★ ★ ★ ★ ★
Conferma aspettative:	★ ★ ★ ★
Design:	★ ★ ★
Affidabilità:	★
Tecnologia:	★ ★ ★ ★ ★
Documentazione:	★ ★
Prezzo/prestazioni:	★ ★ ★ ★ ★

Che cos'è: un campionatore stereo a 16 bit con interfaccia MIDI di serie. Con Amiga accelerati consente di campionare a frequenze superiori ai 50 KHz.

Cosa ci è piaciuto: l'eccellente qualità dell'audio a 16 bit. La facilità d'uso. Il prezzo contenuto. L'estrema semplicità con cui si possono assegnare i campionamenti a una tastiera collegata via MIDI.

Cosa non va: con gli Amiga accelerati (noi abbiamo usato un A1200), che di fatto sono gli unici a poter campionare a frequenze superiori ai 40 KHz in stereo, il software ha dei problemi di funzionamento legati alla funzione fast Amiga. È impossibile il multitasking. La brevità del manuale. Il software potrebbe essere affinato: non è molto stabile e alcune funzioni sono migliorabili.

AMIGA 3D

ARRIVANO NUOVE ESSENZE PER IMAGE

di Antonio De Lorenzo

*Essence è il nome dato
da Steve Worley
alla sua nuova libreria
di tessiture algoritmiche
per Image.
Sessantasei nuove
e preziosissime essenze
per "profumare"
la nostra fantasia 3D*

Nel romanzo *Il Profumo* di Patrick Suskind, considerato uno degli esempi più alti della letteratura di questo secolo e tradotto in 23 lingue (in Italia è pubblicato da Longanesi; è disponibile anche nella collana economica Tea), il protagonista Grenouille nasce privo di qualsiasi odore, ma con un senso dell'olfatto sviluppatissimo, a tal punto da riuscire in pochi anni a elaborare tutti i profumi, gli effluvi e le essenze, diventando prima uno dei più grandi profumieri del mondo e poi perseguendo l'idea di darsi un profumo tutto suo, quell'odore umano che la Natura gli ha così crudelmente negato. Porta avanti questo progetto assassinando decine di giovani fanciulle in modo da sottrarre loro con complesse operazioni gli ingredienti basilari per l'elaborazione di un'essenza umana sintesi di tutte le migliori, assoluta e perfetta. Il romanzo possiede numerosissimi piani di lettura (con un finale sconvolgente) uno dei quali risiede nella metafora dell'artista, di qualunque artista che necessita di un insieme di strumenti basilari per creare la sua arte arrivando a sacrificare se stesso e sconfiggendo spesso anche fuori dalle convenzioni sociali e nell'illegalità pur di perseguire il suo fine (la perfezione).

In grafica 3D, non disponiamo ancora della possibilità d'introdurre sensazioni olfattive (per quanto vi possa risultare strano si sta lavorando persino in questa direzione), ma le texture matematiche costituiscono degli elementi fondamentali per conferire un determinato aspetto agli oggetti, per aggiungere dettagli non altrimenti conferibili. Proprio come per le essenze profumate di cui sopra, è possibile miscelare e inventare commistioni variegatissime al servizio della propria arte grafica.

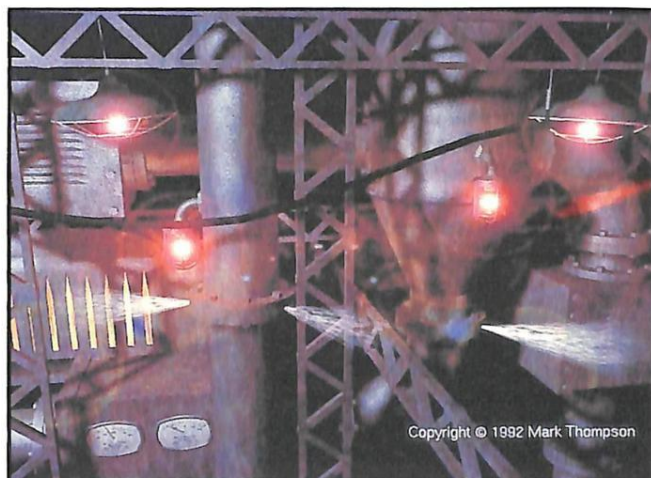
Le Texture algoritmiche

Le Texture algoritmiche o matematiche costituiscono dei preziosi sottoprogrammi richiamati da *Image* all'interno del Detail Editor nel riquadro Attributes per aumentare il dettaglio e la definizione di superficie dei modelli 3D. Le Texture sono quindi degli algoritmi parametrizzabili (l'utente può specificarne oltre ai colori tutta una serie di parametri interni), liberamente posizionabili, delimitabili nell'estensione e anche combinabili sia tra loro (fino a un massimo di quattro texture

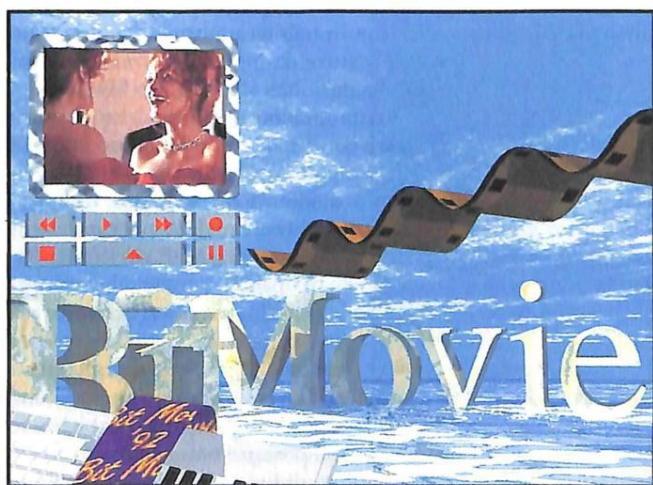
Bit Movie Art



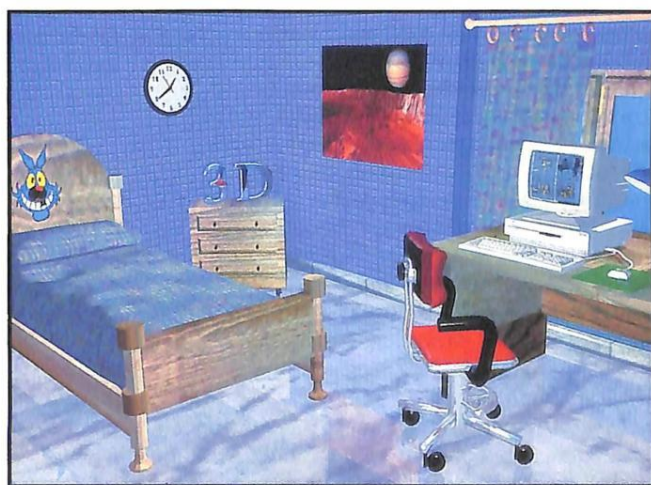
"Picnic" di Eva Cortese (Imagine 2.0)



Anim frame da **"Furnace"** di M. Thompson (LightWave 3D 2.0)



"Bit Movie '92" di Daniele Ficini (Imagine 2.0)



"Room" di S. Epifani e G. Sciarra (Imagine 2.0)

Le immagini presenti in questa pagina sono state gentilmente fornite dal Circolo Ratataplan di cultura Informatica e audiovisiva di Riccione. Ogni anno viene indetta ed organizzata dal circolo, in collaborazione col comune di Riccione, una mostra internazionale di computerarte. Le immagini pubblicate in questa pagina sono state scelte tra quelle inviate dai partecipanti della scorsa edizione nella sezione "Immagine statica Amiga". L'edizione di quest'anno si è tenuta nel periodo 8-12 aprile presso il Palazzo del Turismo. Il bando di concorso è stato pubblicato nello scorso numero, mentre sul prossimo potrete trovare un ampio resoconto della manifestazione. Chi desiderasse ricevere maggiori notizie o delucidazioni può rivolgersi direttamente al direttore artistico: Carlo Mainardi, Via Bologna 13, 47036 Riccione, Tel. e fax 0541/646635. Le immagini a 24 bit proposte in questa pagina sono state stampate su diapositiva dalla Grafic Delta, Via Marecchiese 273, Rimini (FO), Tel. 0541/727868.

contemporaneamente), sia insieme a un massimo di quattro brush. Data la gestione in stati multipli sovrapposti (layer), gli andamenti e i motivi peculiari di ogni texture (sia variazioni della medesima texture, sia assortimenti vari di texture diverse) possono dar luogo a un numero infinito di variazioni, consentendo una scelta estremamente ricca e variegata. Le 14 tessiture fornite finora insieme al pacchetto della Impulse coprono più o meno esaurientemente molte delle esigenze riscontrabili dall'utente, ma non possono naturalmente esaurirle.

Più volte la Impulse ha annunciato la commercializzazione separata di proprie librerie, arrivando addirittura a preannunciare un Texture Editor per la creazione di tessiture personali, ma finora ben pochi passi avanti sono stati fatti in questa direzione. Questo, unito all'assoluta reticenza nel rendere pubbliche le modalità di programmazione (non sappiamo come abbia fatto Steve Worley a strapparle ai programmatori della Impulse, ma senz'altro avrà dovuto stringere un qualche tipo di accordo per entrarne in possesso), ha portato a una situazione di stallo nel loro sviluppo durato diversi anni (ci fu un primo pallido tentativo di comunicazione durante l'ultimo upgrade di *Turbo Silver*, siglato 3.01 SV, predecessore di *Imagine*). Era necessario accontentarsi di quello che la Impulse passava, cercando di sopperire alle mancanze di varietà con la tecnica sorella di brush mapping indagata approfonditamente da *Commodore Gazette* nei numeri scorsi. Nel giugno dello scorso anno, però, Steve Worley, conosciuto per la sua mailing list su rete Ethernet (la rete universitaria USA che collega banche dati di vario tipo) dedicata a *Imagine*, e per il preziosissimo manuale *Understanding Imagine 2.0* (recensito sul numero 9/92 di *Commodore Gazette*), annunciò l'imminente rilascio di una nutritissima ed estesa libreria di tessiture algoritmiche per *Imagine* programmate insieme a Glenn Lewis, autore del package di utility TTDDD esaminato approfonditamente nello scorso numero. Inutile dire che la notizia mise in fibrillazione gli utenti di *Imagine* di mezzo mondo. Ora finalmente il pacchetto è arrivato in Italia, viene importato a un prezzo onesto da un nuovo distributore di Roma e aspetta solo che noi ne scopriamo le infinite applicazioni.

Dotazione

Essence Vol. 1 è privo di contenitore

cartonato o altro packaging costoso. La dotazione è limitata a una spartana confezione composta da semplice cellophane che racchiude un elegante manualletto spiraleto di 116 pagine e un unico disco. Sono presenti ben due cartoline: una di registrazione e una di Follow-up da compilare dopo aver utilizzato il prodotto per riportare impressioni e critiche, oltre che suggerimenti per la creazione di nuove tessiture. Sul dischetto sono presenti 66 tessiture diverse raggruppate in otto directory. Tutte le texture richiedono necessariamente la versione floating point di *Imagine* dalla 0.9 alla 2.0, questo significa che chi non possiede macchine accelerate dotate di processore matematico non può utilizzarle (ricordiamo che l'A4000 montando un 68040 possiede il processore matematico, o FPU, all'interno della CPU). Le tessiture funzionano anche su *Turbo Silver* sempre in versione FP. Dal momento che le varie tessiture occupano dai 5 ai 20K, a seconda del tipo, i 790K totali di codice trovano più agevole sistemazione su hard disk. Il programma non è dotato di utility di Install, in quanto per riportare il tutto su disco rigido è sufficiente copiare le texture nella directory omonima per *Imagine* o *Silver* (Texture).

Texture frattali

La maggior parte delle texture presenti nel pacchetto sono basate su un interessante algoritmo conosciuto col nome di Fractal Noise, che regola la generazione di numeri casuali che possono essere raggruppati in base alla frequenza (o se preferite a quanto velocemente cambiano nel loro valore) e alla scala che descrive nello spazio la loro frequenza di cambiamento. I numeri generati con questo algoritmo vengono fatti corrispondere a colori, consentendo una varietà di effetti e un realismo straordinario con un controllo molto spinto delle caratteristiche di ogni texture. Le texture posseggono anche due parametri temporali (durata e scala) che stabiliscono i tempi del loro mutamento in un certo lasso di tempo (riconducibile al numero di frame desiderato). Il significato di questi ultimi due parametri è presto spiegato. La maggior parte delle texture è stata concepita e si presta per l'algoritmo di generazione a essere implementata dinamicamente, all'interno cioè di animazioni complesse. L'animazione tramite texture algoritmiche è chiamata Texture Morphing. Al

momento, non è possibile trasformare in maniera fluida due texture tipologicamente diverse (e sinceramente non ne vedo minimamente l'utilità). Il morphing vale per texture del medesimo tipo ed è facilmente ottenuto suddividendo la differenza dei parametri delle due texture per il numero di frame impostato. È così possibile con le texture di *Essence* simulare l'atmosfera di un pianeta gassoso in divenire, così come l'ascesa e l'espandersi di volute di fumo o ancora il rincorrersi e trasformarsi delle nuvole.

Dal momento che le varie versioni di *Silver* e di *Imagine* (0.1, 1.1, 2.0) mantengono la compatibilità nella gestione delle texture, ma presentano delle differenze significative d'implementazione, tutte le texture di *Essence* posseggono quello che la casa stessa definisce un "Apex-intelliheader" capace di determinare il tipo di codice richiesto dalle varie versioni. La versione 2.0 di *Imagine* (in attesa dell'imminente release 3) è naturalmente la più avanzata e tra l'altro possiede la prerogativa di visualizzare il colore impostato per la texture in uno spazio del riquadro di definizione. Inoltre, l'ultima versione richiama valori preimpostati per default ed effettua in maniera automatica il rescale della texture quando subentrano modifiche dimensionali sull'oggetto 3D sul quale questa è stata applicata. Le texture fornite con *Essence*, però, hanno caratteristiche aggiunte molto più avanzate rispetto a quelle di provenienza Impulse. Quasi tutte le texture di Worley hanno in più un parametro di FADE per controllare il livello di opacizzazione della texture. È possibile persino disabilitarne i colori se si necessita di sovrapporre diverse e vedere così più agevolmente gli strati sottostanti (layer). Ma la trovata che rasenta la genialità è un'altra. Sebbene le texture non possano naturalmente avvertire l'utente di un'eventuale applicazione scorretta cercando di rimediare all'errore con un messaggio testuale o sonoro, sono in grado di avvertire l'operatore se questo incorre in gravi imprecisioni. L'effetto di una tessitura è limitato a riprodurre il motivo desiderato dall'utente e allora Steve Worley ha pensato bene di utilizzare scacchi di colori diversi a seconda che l'utente incorra in uno dei quattro possibili errori di utilizzo. Osservando i colori che compongono gli scacchi e consultando il manuale, l'utente può così risalire ai valori errati impostati nella definizione della texture. Ma entriamo nel vivo della prova ed esaminiamo ora tutte le texture secondo la loro suddivisione in directory.

Altitude Texture

Otto delle migliori tessiture fornite. Appartengono a questo gruppo quelle texture che conferiscono un disegno variabile ma sempre con un'apparenza in bassorilievo (bumping). Le texture non intervengono sulla configurazione fisica dell'oggetto, ma solo sull'aspetto finale della loro superficie. In altre parole, triangoli, lati e punti non ne vengono influenzati, le texture simulano l'aspetto risultante della superficie. Quello che cambia è il comportamento della luce nell'incontrare le superfici sottoposte a texturing e le ombre conseguenti e variabili a seconda della posizione della sorgente luminosa (che è quindi una variabile critica da considerare attentamente per avere un buon effetto di rilievo).

- **BUMP**: perturba le superfici rendendole composte irregolarmente come può apparire quella di un asteroide o di un terreno, ma è possibile simulare innumerevoli materiali, da fogli di carta a oggetti in metallo battuto.

- **DIAMOND DECK**: simula l'aspetto tipico di molte coperture di scale e atri antiscivolo.

- **FLATTEN**: si tratta di una texture dall'effetto molto strano. Questa texture ombreggia un oggetto per farlo apparire completamente piatto. Se avete per esempio un personaggio, 3D Flatten può essere utilizzata per trasformarlo nell'equivalente 2D e poi simularne l'uscita dalla pagina con un texture morphing che s'incarica di calcolare il passaggio tra i due Universi.

- **SHERESHAD**: è consigliata per applicazioni su sfere per simulare ombre perfette e non poligonali come accade anche nel caso di utilizzo della sfera matematica.

- **HEXMESH**, **GRIDMESH** e **STRIPES**: simulano aspetti regolari e più o meno in rilievo di motivi geometrici per piastrellizzazioni. Rispettivamente con piastrellizzazione esagonale, quadrata e rigata.

- **TREEBARK**: è un'altra ottima implementazione. L'aspetto turbolento che la texture riproduce fa sì che l'aspetto esterno dell'oggetto assuma sembianze

vegetali o ancora d'insetti.

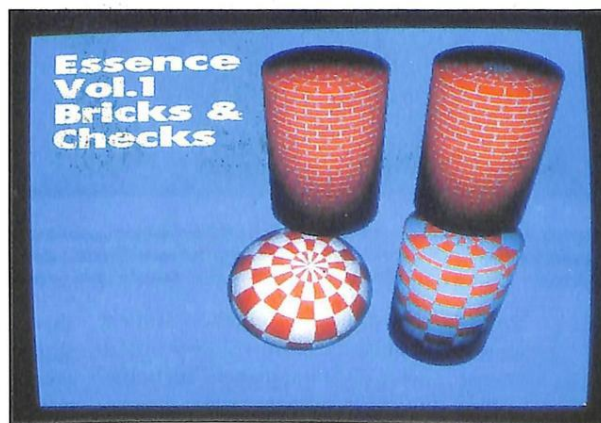
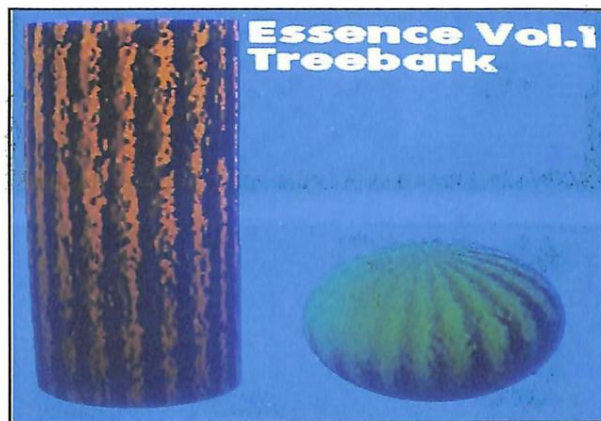
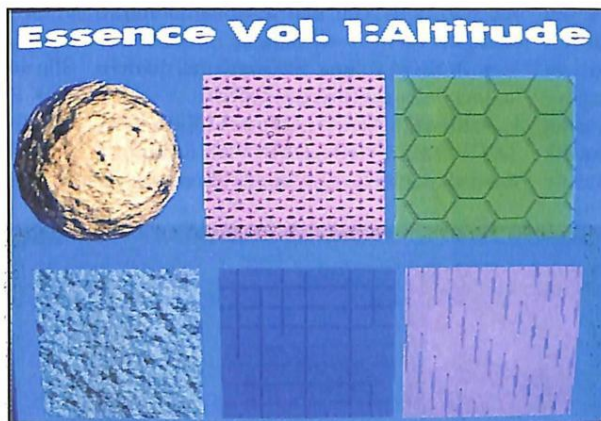


Figura 1: Altitude Texture; da sinistra, Bump, Diamond Deck, Hexmesh, Gridmesh, Stripes. **Figura 2:** Altitude Texture (Treebark) simula la corteccia di un albero e il motivo di un'anguria. **Figura 3:** Bricks e Checks Texture; in alto, Cylindbricks e Cylindbricks2; in basso, Spherechecks e Cylindchecks

alquanto organiche. È possibile per esempio simulare l'aspetto tipico della corteccia di un albero o la scorza di un'anguria come anche l'aspetto esterno di altri

simulare display riportanti l'ora, altezze o qualsiasi altro dato numerico. È possibile anche animare il tutto muovendo le varie cifre in progressione.

Bricks e Checks Texture

Scacchi e muri costituiscono due categorie di texture implementate classicamente in qualsiasi programma di rendering 3D e la Impulse ne fornisce diverse versioni di entrambe le categorie. La raccolta *Essence*, però, ne migliora l'implementazione consentendo di applicarle correttamente anche a oggetti complessi non sempre perfettamente mappati dalle texture Impulse.

- **CYLINDBRICKS** e **CYLINDBRICKS2**: sono due varianti della texture Brick fornita dalla Impulse, molto utili specialmente se utilizzata con progetti architettonici. Servono per estendere il rivestimento mattonato su superfici irregolarmente curve e cilindriche.

- **SPHERECHECKS** e **CYLINDCHECKS**: sono altre varianti delle texture Check della Impulse. Conferiscono l'aspetto scacchato a superfici sferiche e cilindriche.

Miscellaneous Texture

Questa categoria comprende sei tipologie di texture algoritmiche molto specializzate e complesse. Esaminiamole attentamente.

- **BBCOURT**: è forse la texture più specializzata dell'intera collezione. Applica su un piano il disegno standard di un campo di basket.

- **COUNTER**: altra texture estremamente specializzata anche se di più ampio utilizzo rispetto alla precedente. Riproduce infatti un display a cristalli liquidi (LCD). Dal momento che l'utente ne può specificare le cifre, è possibile introdurre il display su consolle varie per simulare display riportanti l'ora, altezze o qualsiasi altro dato numerico. È possibile anche animare il tutto muovendo le varie cifre in progressione.

- **HUEROTATE**: non è una normale texture che applica un qualche tipo di motivo sulla superficie dell'oggetto. Essa considera il colore già presente e lo manipola. Preleva il colore impostato e lo ruota lungo il diagramma circolare dei colori (il diagramma circolare delle tinte è un modo grafico per ordinare i colori su una superficie circolare, dal blu al magenta e dal rosso al giallo). Dal momento che è possibile specificare in gradi la quantità di rotazione, questa texture è molto utile in animazione per ricreare quel famoso effetto di color-cycling in voga nei programmi di generazione frattale e molto ben implementato in *D-Paint*. La differenza è che in questo caso il colore viene considerato in 24 bit puri. Utile anche per simulare il funzionamento di spie luminose.

- **MANDELBROT**: disegna il famoso insieme di Mandelbrot e ne consente lo zoom per evidenziare dettagli e motivi dell'insieme frattale più famoso del mondo.

- **POLKADOTS**: costituisce il motivo di punti circolari di dimensione variabile distribuiti in maniera casuale.

- **VEINEDMARBLE**: è una delle più richieste e avanzate implementazioni dell'intero pacchetto. Dopo anni d'attesa, è finalmente possibile avere una texture che riproduce con assoluta fedeltà e realismo motivi marmorei. Questa texture frattale, infatti, consente di ricostruire i motivi geometrici tipici del marmo simulandone la formazione reale per deformazione di strati sedimentari sovrapposti. Precedentemente, si era costretti a ricorrere alla texture Wood con scarsi risultati o a impiegare più proficuamente il brush mapping.

Noise Texture

Gruppo costituito da 16 texture tra le più avanzate e complesse. Il loro tempo di calcolo è tra i più elevati, la possibilità di sovrapposizione e di trasformazione dinamica ne fanno degli autentici capolavori d'implementazione.

- **BLOBC, BLOF, BLOBR**: costitui-

scono le texture frattali standard in due colori. Consentono di ripetere i motivi di un cielo nuvoloso, dello sporco, così come di riprodurre il motivo di quarzi e particolari varietà di marmi. La desinenza C, F o R si riferisce a Color, Filter e Reflect.

- **BOZO**: è una versione in quattro colori della texture precedente.

- **FRACTALCOLOR e ALTRE UNDICI TEXTURE DERIVATE**: sono i fiori all'occhiello dell'intera libreria. Gra-

proposito). Rispetto ai brush, però, questi pattern essendo texture algoritmiche non sgranano neanche per ingrandimenti elevati (funziona esattamente come per la grafica bitmap e l'equivalente vettoriale).

- **CUBIST**: pavimentazione a cubi con illusione ottica.

- **FLORTILE**: simula la pavimentazione di molti locali pubblici.

- **HEX**: riproduce una pavimentazione a motivi esagonali.

- **TRIANGLE**: un'altra semplice pavimentazione ottenuta per triangoli equilateri di colore alternato.

Swirls Texture

Questo gruppo comprende texture con motivi a vortice o spiraliformi. Possono essere impiegate per simulare i vortici in movimento dell'acqua e dell'aria, così come la formazione di galassie...

- **SWIRL e SWIRL3**: applicano una texture composta da spirali di colori alternati.

- **SWIRLFRACT e SWIRLTURB**: come le precedenti, ma con spirali perturbate.

Transition Texture

Fondamentalmente, queste texture aggiungono un solo colore sulla superficie dell'oggetto seguendo una geometria per lo più a singola banda. La banda di transizione tra il colore impostato nella texture e quello dell'oggetto può essere definita in

posizione, grandezza così come il passaggio tra i due colori può essere scelto con variazione brusca o più morbidamente per tonalità intermedie. Gli utilizzi sono molteplici; si va dalla simulazione del bandeggio di pianeti gassosi, ai motivi dei corpi trisegmentati degli insetti.

- **BANDSM, RINGSM, SHELSM**: creano bande sfumate regolarmente.

- **BANDFRAC e CINQUE DERIVATE**: lavorano come le precedenti solo che le zone di transizione risultano discontinue.

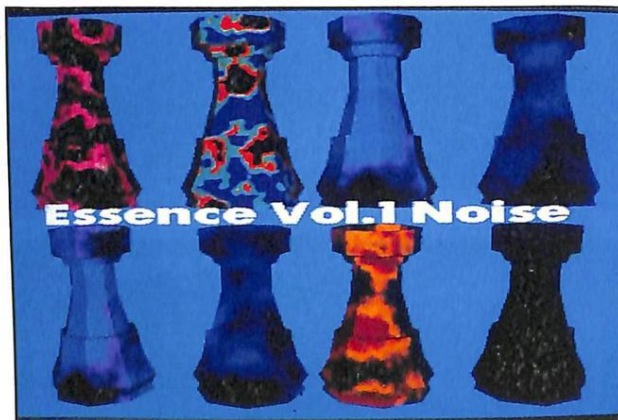
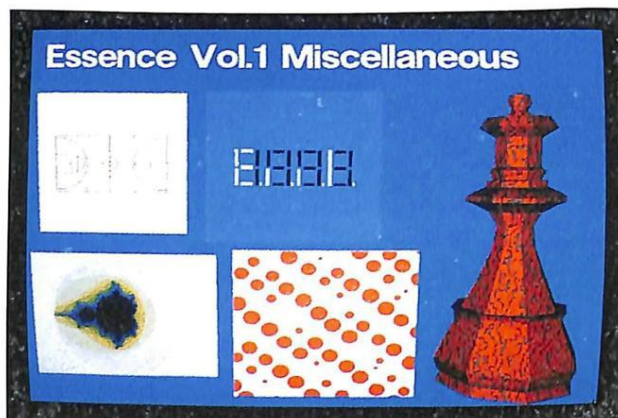


Figura 4: Miscellaneous Texture. Si noti l'estrema varietà e la qualità della texture Veinedmarble applicata sul pezzo degli scacchi. Figura 5: Noise Texture, una tra le più varie

zie alle loro varianti, Filter e Reflect possono simulare cieli nuvolosi in movimento, volute di fumo, pianeti, turbolenze liquide, materiali marmorei...

Pattern Texture

Essence possiede quattro diverse texture per l'introduzione di pattern, vale a dire motivi ripetuti tipici delle pavimentazioni. È molto semplice introdurre questi motivi per brush mapping e tiling (si veda il numero 1/93 di *Commodore Gazette* in

- LINEARISM, CYLINDISM e RADIALISM: funzionano come la texture Linear di *Imagine* con l'aggiunta del controllo per il passaggio di tinta.

- LINEARFRACT e CINQUE DERIVATE: funzionano come le precedenti inserendo una discontinuità specificabile nelle zone di passaggio di tinta.

Utility Texture

L'ultima directory comprende alcune utility che possono risolvere particolari problemi di texture mapping.

- SOLID: applica un colore all'oggetto. Può essere utilizzata per estendere la colorazione a tutti gli oggetti di una gerarchia tramite Apply to children. Dal momento che poi la texture possiede un parametro di Fade, la tinta può essere resa a piacere più attenuata.

- SWAPCRF: consente di scambiare i valori di color, reflect e filter.

- SWAPRGB: molto simile alla precedente consente di miscelare in percentuale componenti di colori.

- VARYABSRGB e VARYRELBRIGHT: permettono d'introdurre variabilità nel colore e nella brillantezza degli oggetti. Molto spesso, infatti, le superfici degli oggetti 3D appaiono perfette, assolutamente regolari e omogenee, ben conformate e con dettagli di superficie troppo netti, perfetti e spesso innaturali e irreali. Queste due texture introducono variabilità, discontinuità di superficie (sovrapponendosi alle caratteristiche di qualsiasi altra texture, brush o attributi presenti) per far sì che gli oggetti da perfetti (come li calcola l'algoritmo di ray tracing) appaiano più o meno imperfetti o degradati (come nella realtà), consentendo di raggiungere maggiore realismo.

Un'occhiata alla documentazione

Dopo un'agile presentazione e una veloce guida d'installazione, il manuale (in inglese) prosegue spiegando l'utilizzo

generale delle Texture in *Imagine* e l'animazione tra esse (Morphing Texture). Segue un capitolo molto ben realizzato sulla matematica delle texture frattali. Ampi spazi vuoti consentono di aggiungere a matita eventuali note o parametri utili riscontrati nelle proprie sperimentazioni (ricordatevi di aggiungere sempre anche le dimensioni dell'oggetto).

Consigliamo vivamente ai nostri lettori l'acquisto dell'originale. Oltre infatti che moralmente giusto, s'incentiva lo svilup-

per altro semplice funzionamento per tentativi frustranti quanto nella maggior parte dei casi del tutto infruttuosi.

Considerazioni, conclusioni, frontiere

Dopo il grande successo riscosso in tutto il mondo col volume *Understanding Imagine 2.0*, Steve Worley mette a segno un altro bel colpo dal momento che nel momento in cui scriviamo il numero di pacchetti venduti ha superato le 10 mila unità. Crediamo di conoscere le ragioni di un così grande successo. Steve Worley è prima di tutto un utente avanzato di *Imagine*, conosce le nostre esigenze e si è scontrato in prima persona con le difficoltà del 3D dal momento che ha iniziato (e prosegue con successo) il suo viaggio nella grafica tridimensionale proprio con *Imagine* della Impulse su Amiga. Tenete conto che collezioni di texture per altri programmi 3D e sistemi operativi sfiorano le decine di milioni di lire e neanche presentano una tale varietà e perfezione. Senza altro Worley svilupperà il secondo volume di texture attualmente in fase avanzata di realizzazione. Consiglierei personalmente a Worley d'implementare un planisfero sotto forma di texture in modo da riprodurre e mappare correttamente l'aspetto della terra vista dallo spazio su un geoide (il brush mapping per operazioni simili rivela spesso grosse limitazioni soprattutto in animazione).

Altra texture di cui si avverte la mancanza è la scagliatura tipica di tetti e pesci, del piumaggio, e infine di peli e capelli. Dal momento che poi Worley è un utente avanzato e un attento ricercatore, ci aspettiamo dopo le texture frattali anche altre tipologie di texture attualmente in studio e sviluppo presso laboratori di ricerca. Queste frontiere, che senz'altro vedremo al più presto anche su Amiga, sono le texture evolutive e le texture sviluppate su tecniche di Reazione-Diffusione. Le texture evolutive consistono nell'implementazione di complessi algoritmi riuniti sotto la sigla di A-Life, che consentono di simulare l'evoluzione darwiniana con popolazioni d'im-

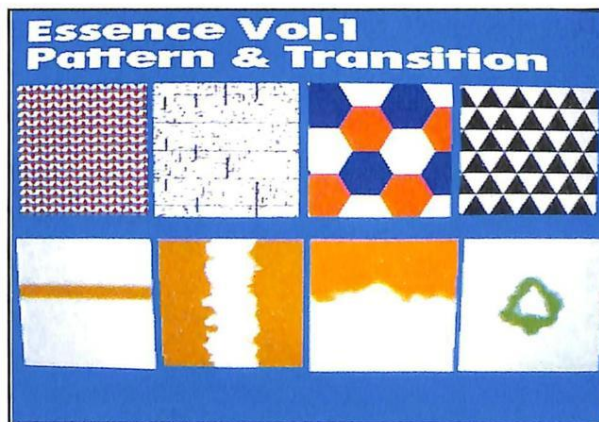
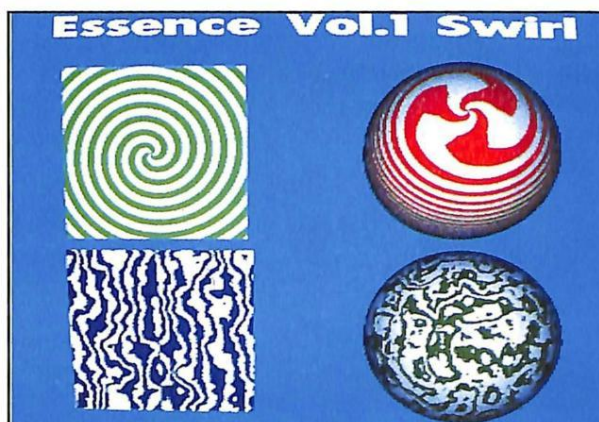


Figura 6: Swirl Texture. Figura 7: Pattern e Transition; fila in alto, Cubist, Flortile, Hex e Triangle della divisione Pattern. In basso, un campione delle numerose Transition Texture

po di altre librerie, è possibile suggerire all'autore l'implementazione di texture delle quali si avverte l'esigenza, si riceve una newsletter con accorgimenti, consigli d'utilizzo e notizie sullo sviluppo di nuovi prodotti. Ora, poi, che è possibile acquistare il tutto in Italia senza rincari ingiustificati, servirsi di una copia pirata è ancor più disonesto e destabilizzante per l'industria del software.

Tenete poi in conto che senza la documentazione l'utilizzo del tutto diviene davvero molto limitato e si perde moltissimo tempo a cercare di capire il

per altro semplice funzionamento per tentativi frustranti quanto nella maggior parte dei casi del tutto infruttuosi.

magini. Vengono selezionate le immagini che presentano caratteristiche estetiche più interessanti e poi introdotte delle mutazioni per generare nuove generazioni di motivi in continuo cambiamento. Un'evoluzione artificiale in luogo di quella naturale mentre l'utente si diverte a ricoprire il ruolo della Natura (o di un Dio, se volete!) introducendo mutazioni e guidando l'evoluzione sui parametri di selezione. Presentate per la prima volta al Prix Ars Electronica 1992 di Linz, Austria, da Andrew Witkin e Micahel Kass, le texture prodotte con tecniche fisiche di Reazione-Diffusione sono prodotte letteralmente per crescita di motivi grafici (pattern) e possiedono sembianze organiche di bellezza e realismo spettacolari. □

Il programma recensito è disponibile presso:

MangaZone Advanced Services
(Essence: L. 145.000 Iva compresa)
Via Grandis, 1
00185 Roma
(Tel./fax 06/7028955)

SCHEDA CRITICA

Prodotto:

ESSENCE 1.0

VOTO:

(In decimi)

9,4

Funzionalità:	★ ★ ★ ★ ★ ★ ★
Conferma aspettative:	★ ★ ★ ★ ★
Affidabilità:	★ ★ ★ ★ ★ ★ ★
Documentazione:	★ ★ ★ ★ ★
Prezzo/prestazioni:	★ ★ ★ ★ ★ ★ ★

Che cos'è: un set di nuove texture algoritmiche per Turbo Silver e Imagine liberamente associabili e mutualmente frammistibili tra loro e con le texture in dotazione con i pacchetti Impulse (Turbo Silver e Imagine).

Cosa ci è piaciuto: quasi tutto. Il prezzo molto conveniente (per prodotti simili su workstation si pagano decine di milioni). La promessa di nuove librerie in uscita basate anche sui desideri dell'utente. La quasi perfetta manualistica. Il supporto della casa con newsletter. I messaggi di errore interni.

Cosa non va: la lentezza delle texture frattali obbliga ad avere macchine accelerate (questo non è un difetto, ma una conseguenza intrinseca alla complessità delle texture implementate). Manca nella documentazione qualche tutorial sull'animazione delle texture frattali, lacuna grave vista la grandissima esperienza e l'incessante attività di Worley nel campo dei tutorial dedicati a Imagine.

INTERVISTA A MIKE HALVORSON

Stare per conoscere il fondatore della Impulse, la software house che ha creato Imagine e la scheda Firecracker

L'intervista che state per leggere è stata fatta via modem e ha coinvolto una quarantina di utenti (tutti utilizzatori provetti di Imagine) collegati contemporaneamente in chat (ossia digitavano le loro domande e ricevevano risposta testuale diretta). Il protagonista è Mike Halvorson, fondatore e direttore della Impulse (fondata nel 1986), programmatore e gran patron della grafica 3D su Amiga. L'intervista risale a un anno fa, ma per questioni di copyright non abbiamo potuto proporvela prima. È stata realizzata grazie alla rete Portal (un servizio statunitense di telecomunicazioni online) ed è stata ritoccata e ripulita da Steve Worley, oltre che introdotta nella mailing list da lui curata dedicata a Imagine presente su Internet. Nonostante risalga a qualche tempo fa, e le risposte risultino a volte molto concise quando non vaghe, contiene notizie di prima mano alquanto interessanti per gli appassionati. In mezzo a un mare di voci, indiscrezioni, bisbigli e pettegolezzi vari, ci aiuterà certo a infilare una serie di punti certi. Ci si renderà meglio conto inoltre del lavoro alacre e incessante condotto dai programmatori sulla base delle richieste dell'utenza registrata. Si avverte inoltre un certo controllo sulla concorrenza e soprattutto non dimentichiamo che alla Impulse i primi utilizzatori sono proprio i programmatori (con lo stesso Mike Halvorson in testa): loro quindi per primi si rendono conto dei limiti e delle possibilità di miglorie. Lascio comunque a voi l'incombenza di trarre le conclusioni e vi auguro buona lettura.

D. La scheda Firecracker 24 di vostra produzione è un buon display video a 24 bit, ma potrebbe essere migliorato. Esiste una Firecracker deinterlacciata o qualche altra scheda in realizzazione? E a quando una scheda compatibile Zorro III che utilizzi memoria in 32 bit per eliminare i wait state o quantunque per proporre un'eccellente piattaforma multimediale?

R. Disponiamo già di una Firecracker 24 a 31 KHz e la memoria a 32 bit è prevista e vi stiamo lavorando. Crediamo che alcuni utenti vedano nella Firecracker più una soluzione ottimale per la stampa che non per il video, così stiamo lavorando al suo miglioramento, così come a un'implementazione diretta del PostScript.

D. Avete piani per lo sviluppo della Firecracker? Caratteristiche come risoluzioni più elevate e processori aggiuntivi? Penso che un processore installato direttamente sulla scheda sia un punto importante.

R. Coprocessore... Si stiamo pensando all'Intel 1860.

D. Che ne è della BBS della Impulse menzionata nella newsletter.

R. Abbiamo appena installato il nostro network.

D. Vedremo in futuro un altitude mapping (bump) più reale?

R. Se conoscete un metodo diverso e migliore rispetto a quello da noi utilizzato, dicitelo...

D. Quando sarà disponibile la versione IBM di Imagine e gli oggetti saranno compatibili con quelli Amiga?

R. Imagine 3.0 per PC e Amiga è ormai in uscita (come sappiamo c'è stato qualche ritardo, ndr). E... sì, i file saranno perfettamente compatibili.

D. Sono molto curioso di sapere come ha iniziato la Impulse, in quanti eravate e in quanti siete ora. E tra cinque anni?

R. Come abbiamo iniziato? Quando come molti altri abbiamo compreso che l'animazione era possibile con un computer che non costasse cifre astronomiche. Al momento siamo in sei alla Impulse e non ho idea di dove saremo fra cinque anni così come tra cinque settimane.

D. Ci sono alcune cose in Imagine che l'utente trova frustranti, come quando il programma richiede se vuoi lasciare il Detail Editor anche se non ci sono oggetti da salvare o non sono state apportate modifiche. La maggior parte del lavoro di sviluppo in Imagine è teso al miglioramento degli algoritmi di rendering e all'introduzione di nuovi effetti o anche l'interfaccia sarà sottoposta a un'accurata revisione?

R. Ciascun utente che utilizzi un programma trova aspetti che non lo soddisfano. Noi cerchiamo di venire incontro e di accontentare il maggior numero di utenti. Se trovi qualcosa che ritieni terribile, documentala, inviacela per iscritto e noi faremo di tutto per rimediare. Il software è in continuo cambiamento. Speriamo di renderlo sempre più facile, veloce e migliore in ciò che gli utenti ci chiedono e tutto questo per una manciata di dollari e magari... in soli cinque giorni. Appreziamo le critiche costruttive, finalizzate. Fateci sapere cosa volete e noi

faremo sempre del nostro meglio per venirci incontro.

D. Quanto pensi costerà il Rendering Engine (letteralmente: motore di rendering)?

R. Se ti riferisci al Rendering Engine accennato nella nostra newsletter è troppo presto per parlarne ora. Posso dire che stiamo lavorando per renderlo affidabile.

D. Il Cycle Editor di Image è una delle sue parti più potenti tanto che personalmente mi piace moltissimo. Tuttavia, come sicuramente sai, è abbastanza difficile utilizzare segmenti rigidi per modellare curve uniformi che si piegano altrettanto difficilmente. Veniamo allora alla mia domanda: ci sono previsioni per l'introduzione di oggetti basati su spline nelle prossime versioni di Image?

R. Le spline richiedono quantità di memoria non indifferenti. Stiamo lavorando a una forma di spline che renderà possibile ciò che desideri in un batter d'occhio. La risposta è quindi sì.

D. Che supporto avrà Image per lo standard SAGE o la scheda Resolver? La versione per IBM supporterà il processore Intel 860 direttamente?

R. Quando avremo modo di vedere ed esaminare i prodotti che aderiscono allo standard SAGE o le caratteristiche della Resolver saremo in posizione di valutarne le proprietà. Al momento, ancora non abbiamo visto tali prodotti e odio fare commenti su qualcosa che non si è ancora materializzato in un prodotto reale (dalle newsletter successive apprendiamo che lo standard ha suscitato un'accoglienza piuttosto tiepida da parte dei programmatori Impulse, ndr).

D. Ho letto che Scapemaker crea elevazioni basate su valori RGB di un'immagine IFF. Perché allora il requester dei colori in Image non potrebbe fare lo stesso con i check-box e gli slider?

R. Non capisco la domanda.

D. Il requester degli attributi nel Detail Editor possiede tre slider e alcuni check-box (selettori per l'attivazione di operazioni, ndr) per applicare i colori, ma i requester di colori delle texture sono diversi. Perché?

R. Abbiamo aggiunto il box del colore nel riquadro delle Texture solo all'ultimo momento perché Louis Markoya lo riteneva assolutamente indispensabile (e vorrei vedere, ndr). Nella prossima versione intendiamo inserire maggiore continuità tra i due riquadri. Prendiamo nota della tua segnalazione.

D. Non è prevista nessuna versione della Firecracker 24 esterna per i possessori di un A500?

R. Mi dispiace ma la risposta è negativa, ma puoi acquistare una Bodega Bay (è un'espansione per A500 che consente di alloggiare al suo interno le schede per l'A2000, ndr) che funziona egregiamente.

D. Non per suscitare le tue ire, ma dal momento che Image possiede una curva di apprendimento così ripida, sarebbe davvero auspicabile avere un manuale decente. Un pacchetto viene pagato in upgrade 100 dollari solo se migliore della versione precedente. Fate scrivere a Steve Worley il prossimo!

R. Grazie per il tuo suggerimento. Comunque, non sono convinto che Worley possa fare giustizia definitiva, lui scrive grandi tutorial, ma c'è di più in un software. Steve ha fatto un grande lavoro con il suo recente libro (*Understanding Image 2.0*, recensito sul numero 9/92, ndr), ma stiamo ancora cercando qualcuno che possa fare il miglior lavoro possibile. Spero solo di portare il software a uno stadio in cui non ci sarà bisogno di un manuale, questa è un'idea migliore. Il costo non ha importanza.

D. Siete consapevoli che in modalità Cylindric Mapping Image non esegue la ripetizione del brush come già indicatovi in una mia lettera. Non ho ricevuto risposta alla mia segnalazione e la nuova versione 2.0 non ha risolto il problema.

R. Non ancora, ma vi guarderò personalmente.

D. Più di una domanda vorrei fare un commento/richiesta. Trovo che le texture algoritmiche (Check, Wood...) siano una grande implementazione e spero di vederne in quantità maggiore. Allora perché non rilasciare il formato di codifica in modo tale che gli utenti possano scriverne di proprio conto e magari metterle a disposizione degli altri utenti?

R. Ottima idea Steve, ma sono molto difficili da compilare e noi non siamo preparati a supportare le molteplici domande che potreste rivolgerci su come realizzarle. Può darsi che in futuro le renderemo accessibili tramite ARexx facendovi impazzire dalle possibilità.

D. Bene, vedremo il morphing e i sistemi di particelle, il vero ray tracing e altre

interessanti caratteristiche come queste. Ma qual è la caratteristica più prossima sulla quale state lavorando?

R. Sincronizzazione e tracking di suono e musica in animazione.

D. Due richieste veloci che spero considerate d'implementare nella prossima release di Image: rendere i progetti più "trasportabili" in modo che copiando il progetto su un altro disco altri utenti possano caricarli come sono, e ingrandire il file requester.

R. OK.

D. La nostra attenzione di grafici 3D questi giorni è catturata dal morphing (alla Terminator 2, per intenderci). Pensi che Image crescerà verso questa direzione? Che ci dici del "character animation system"? Le due cose lavoreranno congiuntamente?

R. Sì. Il morphing sarà migliorato e il Cycle Editor diventerà molto più potente e versatile in futuro.

D. A proposito di rendere i progetti più trasportabili, c'è la possibilità che in un prossimo futuro il programma possa leggere direttamente i file DEM?

R. Qualcuno può dirmi cosa sono i file DEM?

D. Digital Elevation Map (i file di codifica 3D del territorio introdotti da Vista Pro della Virtual Reality Lab letti anche da Scenery Animator, ndr).

R. Penso di no. Ma potrebbero esserci maggiori sforzi per rendere più vicine le interfacce di Vista Pro e Image (sappiamo di collaborazioni incrociate tra i due team di programmatori, ndr).

D. Ci sono piani per implementare un algoritmo di morphing generale in Image o posso continuare a lavorare per mio conto con TTDDLIB? Per morphing generale intendo un morphing svincolato dalla geometria o topologia dei due oggetti che possono pertanto possedere un numero diverso di punti, lati e facce con differenti caratteristiche di colorazione, riflessione, trasparenza... (la domanda è stata posta da Glenn Lewis autore della libreria TT3DLIB e coautore insieme a Steve Worley di Essence).

R. Abbiamo ancora un problema con oggetti aventi un numero di punti diversi. Le altre proprietà funzionano bene. Sì, credo che risolveremo tutti i problemi legati al morphing.

D. Ci sarà la possibilità nella prossima versione di scegliere una modalità di calcolo intermedia tra Scanline e Trace che consenta le ombre, ma non riflessioni e rifrazioni? Ho espresso la richiesta per lettera come suggerisci di fare, ma ho l'impressione che non l'abbiate letta.

R. La prossima versione avrà una possibilità come quella da te suggerita. E altro ancora naturalmente.

D. Due brevi domande: sono disponibili informazioni tecniche sul formato di salvataggio dell'Action Editor? Sarà presente un comando per interrompere e riprendere il rendering?

R. Le risposte alle tue domande sono no per la prima e sì per la seconda.

D. Amo la possibilità di nebulizzare gli oggetti (Fog). In futuro gli oggetti nebulizzati potranno ricevere e produrre anche ombre?

R. Gli oggetti nebulizzati sono così difficili da realizzare e gestire che dubito potranno fare quanto suggerisci, ma sto parlando dello stato attuale dei lavori e modifiche. Non penso comunque che la prossima versione comprenderà quanto da te richiesto.

D. Una piccola richiesta. Utilizzo il comando Camera Retrack nello Stage Editor, anziché ogni volta digitare il nome dell'oggetto su cui puntare la camera, è possibile avere l'ultimo nome introdotto automaticamente all'interno del requester? Sarebbe alquanto più veloce...

R. Ottima idea, lo faremo. Chissà perché non ci avevo pensato prima.

D. Avete in progetto di aggiungere animazioni ambientali calcolate con Vista come sfondi nelle animazioni con Image? Per esempio un paesaggio montano in movimento (generato con Vista) con sovrapposto un aeroplano (generato da Image).

R. Questo è già possibile con Vista Pro e Image. Ho appena terminato un'animazione di 12 mila fotogrammi usando Vista Pro 2.0.

D. È in progetto l'introduzione di un Alpha Channel per la Firecracker 24?

R. Non nella versione presente. Può essere in futuro.

ANTEPRIME 3D

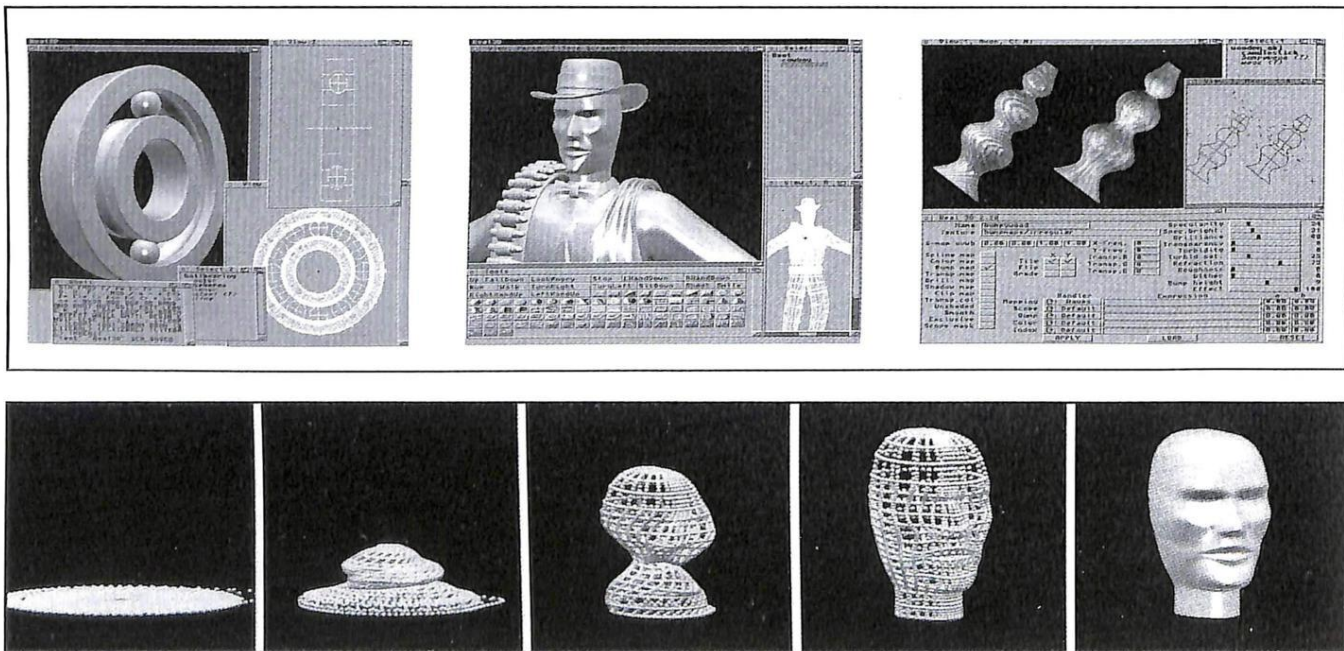
REAL 3D 2.22

Le principali novità della nuova versione del programma della Activa

A d Hannover si è tenuta l'ennesima edizione del CeBIT, la più importante fiera europea dedicata al mondo dell'informatica. Le interessantissime novità presentate dalla Commodore, Amiga 4000 Tower e A4091 (controller SCSI 2 per Amiga 3000 e 4000), erano accompagnate da numerose soluzioni hardware/software nell'ambito della videografia bidimensionale e tridimensionale. All'interno del grande stand della Commodore

Da tempo, circolavano voci che questa versione del programma avrebbe rivoluzionato il mondo della grafica tridimensionale e le caratteristiche elencate nel comunicato stampa apparso a novembre avevano creato un'attesa a dir poco spasmodica da parte dei professionisti. Si cominciò a parlare di multitasking reale in fase di editing (per esempio poter modificare oggetti mentre il programma è in fase di rendering), di applicazioni di

Quello che ho visto fare al CeBIT ha dell'incredibile: forse avrete già sentito parlare d'"intelligenza" degli oggetti componenti una scena; per intelligenza è intesa una sorta di personalità dell'oggetto. Ognuno di questi può infatti avvisare il programma sulla sua consistenza fisica, sulle leggi che lo vincolano all'interno di un sistema fisico e sulle reazioni che deve produrre a contatto con altri oggetti "intelligenti". Ora pensate di dover disegnare un'animazione complessa nella quale alcuni oggetti di forme diverse vengono lasciati cadere da una scalinata. Il metodo finora disponibile era creare un percorso per ogni singolo oggetto, cercando di simulare nella maniera più realistica possibile la forza di gravità, il rimbalzo dell'oggetto sulla superficie dello scalino. Quest'operazione può richiedere diverse ore di lavorazione sulla scena. Per un attimo sognate di poter assegnare ai singoli oggetti un peso speci-



Sopra: la nuova interfaccia grafica di Real 3D 2.22 della Activa è completamente riconfigurabile (il programma è importato in Italia dalla AP&S di Udine). Sotto: le sferette disposte in modo casuale vengono a disegnare la forma della testa con un movimento circolare

una fiumana di gente ammirava con entusiasmo la presentazione di *Real 3D* nella sua seconda versione, interamente riscritta. Prima ancora della sua effettiva commercializzazione si è già giunti alla versione 2.22, dove sono state aggiunte diverse funzioni e potenziate le già ottime caratteristiche. Devo personalmente ammettere di essere rimasto a bocca aperta per quello che ho visto fare dagli operatori nelle loro dimostrazioni.

leggi fisiche per la descrizione delle scene, di generatori frattali per la creazione di paesaggi e alberi, di animazioni gerarchiche complesse. Le notizie sparse dai pochi fortunati che realmente l'avevano visto in funzione, non fecero altro che impazientire gli appassionati.... sappiate comunque che manca veramente poco alla commercializzazione effettiva del programma, che in Italia viene importato dall'AP&S di Udine (Tel./fax 0432/759264).

fico, un baricentro, una consistenza: lasciateli scivolare giù dagli scalini e automaticamente guardate gli oggetti rimbalzare in maniera perfetta... Con *Real 3D* la gravità, le collisioni, l'elasticità degli oggetti sono tutte calcolate in modo automatico dal programma.

Un altro esempio molto riuscito è quello del bowling, dove i birilli cadono al contatto con la bocca. Quando si dice che ogni oggetto ha una sua consistenza si

pensa subito a un oggetto solido: già si vede la possibilità di "spaccare" vetrate con scritte, bicchieri che al contatto con il terreno si frantumano in mille pezzi. Forse si riesce anche a immaginare la reazione di una palla lanciata contro il muro, con la sua compressione elastica al momento dell'impatto. Sembra invece impossibile dare a un oggetto la consistenza per esempio dell'acqua: non sto parlando di rifrazione dell'oggetto, ma di forma e caratteristica fisica. Riuscirete a stento a crederci, ma *Real 3D* offre la possibilità di giocare con l'acqua come se fosse veramente reale! Immaginate quindi di rovesciare del liquido all'interno di una tubatura... l'acqua che scorre, rallenta dove il tubo curva, per poi fuoriuscire dalla parte esterna del percorso. Forse l'esempio non è chiaro: create un bicchiere, riempitelo con un oggetto che abbia la consistenza dell'acqua; ora agitate il bicchiere e ammirate il risultato. L'oggetto "acqua" assumerà una forma irregolare fuoriuscendo in maniera del tutto realistica.

Se poi si pensa che è possibile creare paesaggi tridimensionali grazie ai generatori frattali, risulta estremamente facile creare situazioni del tutto simili alla realtà: il metodo è molto simile ai programmi *VistaPro 3* e *Scenery Animator*, la

descrizione della scena è creata artificialmente in base a dei parametri impostati nel programma. Questo poi si occuperà di calcolare la consistenza del terreno a seconda dell'altitudine (erba, roccia, neve) o la presenza di laghi, fiumi e alberi. Gli alberi vengono creati come oggetti, che possono essere modificati nell'editor a seconda delle esigenze. Quelli che venivano mostrati al CeBIT erano interessanti per la presenza di foglie molto particolareggiate. La cosa veramente piacevole di questo programma è la possibilità di essere pilotato da script RPL in formato ASCII: questo vuole dire che si possono facilmente inserire nuovi algoritmi di calcolo frattale, come formule matematiche complesse. Torniamo ora alla possibilità di "personalizzare" gli oggetti. Bene, pensate di descrivere un paesaggio virtuale, per quanto realistico possa essere, risulta troppo statico, potremmo dire freddo. Inseriamo la formula del vento e tutto cambierà, dando vita a situazioni particolarmente interessanti come il movimento delle foglie, lo spostamento delle nuvole.

Se quello che ho descritto finora non vi sembra ancora troppo eccitante, devo pensare che siete abituati veramente bene. Davvero quello ho detto non vi

basta? Allora proviamo a mettere una pentola piena d'acqua su un fornello, accendiamo il fuoco e guardiamo le reazioni nell'acqua: tante piccole bollicine si formeranno sul fondo del contenitore e piano piano creeranno la sensazione di una reale ebollizione del liquido. Con i nuovi tool descriviamo la testa di un personaggio, operazione resa molto semplice grazie all'utilizzo di curve B-spline. Riempiamo ora la scena di tante piccole sfere e impostiamo l'animazione affinché queste vengano a disegnare la forma della testa: nell'esempio visto, le sferette, disposte in modo sparso sulla scena, incominceranno a ruotare e con un moto a spirale formeranno la testa precedentemente disegnata. Ora vi sembrerà che nulla sia impossibile con *Real 3D 2.22*, ma ovviamente le sue limitazioni il programma le ha ancora, quando dico che il rendering è veloce, non aspettatevi di vedere generare le scene in tempo reale, o di pilotare il programma con impulsi cerebrali facendo calcolare tutto a lui: *Real 3D* è un grande programma, ma richiede sempre un certo impegno. Sicuramente, comunque, ha fatto passi da gigante e introdurrà definitivamente l'Amiga nel campo della grafica professionale. (a cura di Marco Dufour) □

DIETRO ALL'IMMAGINE

Come nasce un'immagine 3D particolarmente riuscita. Storia e descrizione operativa di "FERRARI F40"

Ritorna la rubrica dopo un numero di assenza. Questa volta prendiamo in considerazione un'immagine realizzata con *Sculpt Animate 4D* della Byte by Byte (dopo aver ospitato resoconti d'immagini ottenute con *Imagine* e *Caligari*). Sebbene infatti l'immagine non raggiunga il top del fotorealismo, è stata realizzata con molta passione e dedizione con un programma alquanto limitato in rendering (meno in animazione). Nonostante abbia qualche anno sulle spalle, ci sentiamo ancora di consigliare il pacchetto a chi si avvicina per la prima volta alla grafica 3D. *Sculpt Animate 4D* esiste ora solo in versione Macintosh e comunque anche in quell'ambiente viene utilizzato prevalentemente come ottimo modellatore. La scelta poi di questa immagine è doppiamente affettiva. *Sculpt Animate 4D* ha

aperto per primo insieme a *Videoscape*, l'universo 3D su Amiga (e molti amighisti si sono formati con questo programma, tra cui molti utenti poi sono passati ad altri sistemi serbandone comunque un buon ricordo). E poi come ammiratori della Ferrari, la casa del cavallino rampante che ha reso le macchine di Maranello un mito sempre attuale in tutto il mondo.

L'autore

Tiziano Diamanti, autore dell'immagine, è un ragazzo di 19 anni che frequenta l'ultimo anno del liceo scientifico. Abita a Monte Vidon Corrado, un piccolo paese in provincia di Ascoli Piceno e ci fa spiritosamente notare quanto in questo

centro sia "difficile trovare appassionati di computer, figuriamoci poi di grafica 3D!". Nostro affezionato lettore da lunga data, Tiziano ha iniziato la sua "carriera" nel lontano (in termini informatici) Natale del 1985 quando si fece regalare un economico C-16 con il quale ha imparato autodidatticamente a programmare in Basic. Dopo quattro anni di frustrazioni e amarezze decise di regalarsi (questa volta pagando lui!) un computer di cui non si pentì: un Amiga 500 fresco di fabbrica. I suoi inizi ricalcano quelli di molti appassionati. Per lo più giocherellava con i magnifici videogame disponibili, incredibilmente affascinato dalla qualità degli stessi e dal loro sonoro (soprattutto in confronto alle caratteristiche del C-16) e poi sempre più conquistato dalle immagini ray tracing presenti nelle demo che circolavano al tempo e continuavano tutt'ora a circolare nella ricca e variegata comunità Amiga. Ricorda con nostalgia di come si domandò allora vedendo quelle splendide immagini, se un giorno sarebbe riuscito a fare altrettanto anche in vista del fatto che per il disegno 2D si sentiva innegabilmente negato. Conosceva solo *Sculpt Animate 4D* e così nella

primavera del '91 se ne studiò accuratamente la documentazione. Passato per il periodo "delle sfere a specchio su pavimenti a scacchi bianchi e rossi", iniziò quello che può considerare a tutti gli effetti il suo primo progetto: la riproduzione di una Ferrari di Formula 1 dell'anno 1989, la Ferrari modello 640, come tiene a sottolineare.

L'opera

Per la descrizione dell'opera lasciamo la parola direttamente all'autore: «Proprio mentre modellavo il tutto, in TV passavano le animazioni e le immagini di Imagina '91 e naturalmente queste mi stimolarono molto nell'impegno. Scelsi come progetto una Ferrari, perché da sempre appassionato di F1, che seguo sempre televisivamente e quando posso anche dal vivo. Da buon italiano non posso che tifare per le rosse di Maranello!

Per creare la F29A ovviamente sviluppai il progetto già esistente della 640 precedentemente modellata e quindi gran parte del lavoro era già bell'e pronto (in quanto per la 640 sono partito completamente da zero, disegnando quasi ogni punto manualmente e non avvalendomi se non limitatamente dei vari tool automatici di tracciamento). Ritoccai quindi la parte frontale, ora non più a "papera" e rialzata, le "pance" che nella F29A sono staccate dalla plancia e disegnai ex novo il doppio fondo piatto e i logo degli sponsor che nel precedente progetto erano assenti.

La realizzazione dei logo degli sponsor è stata particolarmente difficile. *Sculpt Animate*, infatti, non dispone di operazioni, per quanto limitate, di brush mapping e così ho dovuto percorrere vie traverse per raggiungere il risultato agognato. I logo sono stati infatti disegnati con *D-Paint* e convertiti in oggetti 3D per *Sculpt* per mezzo di un'utilità shareware chiamata *Brush 4D*. Poi sono stati fatti aderire alle rispettive superfici di contatto per evitare di evidenziare la natura di oggetti separati degli stessi. Tra le difficoltà maggiori riscontrate ricordo sicuramente il cercare di mantenere il rispetto delle proporzioni e la fedeltà con l'originale. Ma anche problemi di tipo pratico come la mancanza di memoria e la lentezza del computer (che al tempo non era accelerato). Nonostante infatti avessi espanso il mio A500 a 3 MB totali di memoria, durante il rendering finale, *Sculpt* mi informò che restavano solo 4K liberi! Ora

comunque ho risolto degnamente entrambi i problemi avendo da poco acquistato un favoloso A4000 con il quale sto elaborando diverse nuove immagini avvalendomi però del ben più avanzato e potente *Imagine 2.0* della Impulse. Non ho però intenzione di modellare la nuova F93A anche perché rispetto alla F92A

non ci sono molte differenze (speriamo invece che ce ne siano nei risultati di gara!).»

Nonostante fosse la sua prima vera opera 3D, con questa immagine Tiziano si è piazzato al 5° posto assoluto nel concorso d'immagine statica nell'edizione 1992 del Bit Movie. □



NEWS 3D

Notizie, upgrade, anticipazioni, indiscrezioni e curiosità in 3D

Impulse Graphics World

È il nuovo nome del bollettino Impulse recapitato agli utenti registrati. Diverse le novità annunciate. Per iniziare, la Impulse comunica che è pronta la versione 2.0 di *Imagine* per MS-DOS mentre la versione 3.0 per Amiga vedrà la luce per l'inizio dell'estate. Probabilmente, come la Virtual Reality Labs e il suo *Vista Pro*, la Impulse ha deciso di aggiornare la versione Amiga e lasciare di volta in volta una release di differenza tra il mondo MS-DOS e quello Commodore. Questo significa che la versione 3.0 di *Imagine* sarà disponibile per MS-DOS quando uscirà la

versione 4.0 per Amiga. Nel frattempo, rimane confermato che dato il gran numero di aggiunte si è provveduto a redigere la documentazione con nuovi tutorial e illustrazioni, compreso un indice degno di questo nome (sarà la volta buona per avere una documentazione di prim'ordine?). Come nel passaggio dalla versione 1.0 alla 2.0, anche per la nuova release 3 cambierà l'immagine d'apertura da sempre biglietto da visita spettacolare per questo grande pacchetto. Queste le nuove caratteristiche che la Impulse anticipa (non si tratta comunque dell'elenco completo di tutte le novità, si rimanda il lettore alla prima anticipazione da noi pubblicata sul numero 8/92):

• **BONES.** Scheletro interno degli oggetti. Sarà possibile muovere oggetti contigui senza la necessità di unirli.

• **REALIZZAZIONE IN TEMPO REALE DEI MOVIMENTI ALL'INTERNO DELLO STAGE EDITOR.** Fare animazione non potrà risultare più semplice. In altre parole, l'utente potrà vedere la camera muoversi e nel contempo guardare cosa essa inquadrava durante il movimento. Muovendo la camera sarà possibile creare percorsi fluidi basati su spline (spline-path) con un solo semplice comando.

• **BRUSH AND TEXTURE TACKING.** Finalmente, sarà possibile proiettare brush e tessiture su porzioni precise dell'oggetto in modo che durante il movimento e la riscalatura, o per qualsiasi altra modifica, l'immagine rimanga fissa sull'oggetto. Tack significa infatti "incollare". Al momento, quando si muove un oggetto la texture "cammina" letteralmente sulla sua superficie dal momento che con la versione 2.0 *Imagine* ritaglia la texture dal volume di spazio circostante (texture volumetriche).

• **SOUND TRACKING.** Introduzione, gestione, sincronizzazione di suoni campionati all'interno dell'Action editor per mezzo di timeline.

• **ACTION IN STAGE.** Richiamo della Timeline di Action all'interno dello Stage Editor.

• **ACCESSO** individuale a ogni oggetto componente un gruppo da Stage Editor.

• **INTERFACCIA GRAFICA** per percorso su path (traiettorie).

• **SUPPORTO DEL FORMATO GRAFICO COMPRESSO JPEG.**

• **SHADOWS SCANLINE.** Possibilità d'inserimento delle ombre anche in modalità calcolo Scanline (non obbligatoriamente quindi in ray tracing).

• **ANTIALIASING.** Migliorato in maniera sostanziale l'algoritmo per l'eliminazione delle seghettature, vera croce della versione 2.0.

• **INTRODUZIONE DI NUOVI TOOL DI MODELLAZIONE.** Twist, Taper, Bend, Shear e altre deformazioni

ancora introdotte per prime da *PoNGo* (chiamato *Morphus* negli USA).

• **AGA.** Supporto pieno dell'Ham8 e dei nuovi modi grafici AGA.

• **NEW BUTTON.** Nuova fila di selettori programmabili in basso a ogni editor.

• **MACRO.** Finalmente molti aspetti del programma saranno programmabili e automatizzabili. Sarà forse *ARexx* o un linguaggio interno proprietario?

• **FORMS EDITOR** più avanzato.

• **POST PRODUCTION ANIMATION TOOLS.** Nuovi tool per facilitare animazioni in single frame e riversamenti.

• **NUOVO EDITOR TERRAIN.**

• **TEXTURATOR.**

Delle ultime due aggiunte non sappiamo altro se non la loro denominazione. Il primo comunque dovrebbe essere un editor di paesaggi (come il vecchio e glorioso *Terrain* sempre di produzione Impulse), mentre il secondo dovrebbe contemplare la possibilità di vedere direttamente in un riquadro l'aspetto delle texture a mano a mano che s'introducono parametri diversi.

Tutto il bollettino è all'insegna dell'ilarità e dello scherzo (in un punto si arriva scherzosamente ad affermare che anche il nuovo presidente Bill Clinton utilizza *Imagine* per le sue presentazioni al Congresso degli Stati Uniti). La comunicazione si conclude con la frase: "Rimanete sintonizzati, stiamo per buttare all'aria il vostro parrucchino!". Frase alquanto enigmatica, si tratta di un gioco di parole tra "rimanete sintonizzati" e il "soffiare via il parrucchino", nel senso della forte musica in uscita (da intendersi come la gran mole di novità e aggiunte al programma).

Oltre a queste anticipazioni (se alla Impulse cercano, come ormai tradizione, di creare una fortissima attesa, devo dire ci stanno riuscendo in pieno...), la software house di *Imagine* rilascia qualche altra anticipazione. Sta per essere commercializzato anche *Visionaire* nuovo e avanzato programma di morphing distribuito dalla Impulse, ma realizzato da programmatori brasiliani. Il programma supporterà l'*ARexx* e la Impulse dichiara possibili persino gli effetti di morphing realizzati in *Black and White* di Michael Jackson (full

motion morphing). Durante lo sviluppo di *Imagine* i programmatori e beta tester hanno creato diverse animazioni di prova. È stata effettuata una cernita tra le migliori, che sono state incluse nella prima videocassetta della Impulse chiamata *A Little Video Noise*. Si tratta di 30 minuti di animazioni eseguite con *Imagine*, *Visionaire* e *Morphus* accompagnate da un'ottima colonna sonora. L'intera videocassetta è stata realizzata con la scheda a 24 bit Firecracker e il programma pittorico *Light 24* fornito insieme alla scheda stessa. La trovata è eminentemente pubblicitaria, dal momento che ogni azienda realizza periodicamente videocassette dimostrative che mostrano in maniera spettacolare le possibilità dei propri programmi. L'intera videocassetta è stata realizzata con programmi e apparecchiature liberamente disponibili e acquistabili. La qualità è broadcast e tutte le animazioni sono in single frame, quindi a 24 bit puri (per qualcosa come circa 43 mila fotogrammi totali). È stato utilizzato un registratore a passo uno Panasonic AG7500A con un frame controller interno BCD 5000. Ci si è poi serviti di un transcoder UTP2 per prelevare il segnale RGB dalla Firecracker e convertirlo in video S. Una volta masterizzato il tutto, le animazioni sono state editate grazie a due registratori Panasonic (AG7600 e AG7650). Il sistema era tenuto insieme da un controller serie AG800, sempre Panasonic, e un Digital Switch WJ-MX 50.

La Impulse commercializza due set di dischi di oggetti provenienti da un contratto stipulato con l'americana ViewPoint, che si occupa di digitalizzazione 3D in massima qualità (digitalizzazione laser), ma avverte che, a causa del fatto che il team Impulse è molto occupato nello sviluppo dei vari prodotti e di altri nuovi prodotti (vengono buttati lì soltanto due nomi come *Body Shop* e *The Dream Machine*), non saranno resi disponibili per diverso tempo altri set di oggetti 3D e che occorrerà rivolgersi direttamente alla ViewPoint per le proprie esigenze. Infine, si segnala la disponibilità per i super fanatici di una nuova maglietta con il logo di *Imagine* in versione 3.

Ora qualche prezzo. *Visionaire* costa \$119 (disponibile a \$59 in offerta lancio per un periodo limitato di tempo). L'upgrade di *Imagine 3.0* dalla versione 2.0 costa \$100. Infine, la videocassetta *A Little Video Noise* è venduta al prezzo di \$20 e la t-shirt di *Imagine* a \$15 (Impulse, 8416 Xerxes Ave. North, Minneapolis, Minnesota 55444, USA). □

TECHNO 3D

Trucchi, scorciatoie, costruzione di oggetti sofisticati, bug e quant'altro i manuali d'utenza non riportano.

I lettori che ritengono di aver scoperto accorgimenti, scorciatoie, metodi di lavoro o curiosità di comportamento riguardanti i più svariati programmi 3D possono inviarci il loro materiale. Pubblicheremo i contributi ritenuti più interessanti.

Dopo aver occupato i passati numeri della rubrica con alcuni argomenti monografici (lo ricordiamo per i più distratti, sono state trattate le modalità per ottenere oggetti vitrei, modellare e gestire lenti d'ingrandimento, oltre ad affrontare nello scorso fascicolo le complesse problematiche per la modellazione di un lapis perfetto nelle sue componenti), torniamo con un appuntamento variegato e stracolmo di trucchi e accorgimenti diversi e interessanti per *Real 3D*, *Pixel 3D*, *Imagine* ed *Essence*.

Animazioni con fotogrammi identici ripetuti

L'accorgimento che vi presento è nella sua semplicità molto utile e consente di risparmiare una notevole quantità di tempo riguardante il calcolo necessario per terminare un'animazione. Ogni programma 3D deve infatti procedere a eseguire i calcoli per ciascun fotogramma, anche quando una sequenza più o meno lunga è formata da uno stesso fotogramma ripetuto (perché per esigenze narrative particolari la camera, per esempio, deve fissare una certa porzione della scena per un determinato tempo). Specialmente se il tempo a disposizione è breve, e quello per i calcoli elevato, questo trucco potrà farvi risparmiare ore o addirittura giorni interi di calcolo! L'idea mi è venuta mentre stavo completando la mia animazione "Disavventure di un collezionista" che ho presentato al Bit Movie. Poiché l'animazione è la prima al mondo a utilizzare in maniera intensiva le lenti (lavori su workstation comprese, mai nessuno prima aveva realizzato qualcosa di simile), i tempi di calcolo rimanevano sempre molto elevati, seppure utilizzassi una macchina accelerata via 68030 a 33 MHz e relativo processore matematico (una media di un'ora e mezza per ciascun fotogramma, per quasi 600 frame), men-

tre i termini di scadenza per la partecipazione al Bit Movie si facevano sempre più vicini. Forte del vecchio adagio "la necessità aguzza l'ingegno", ho pensato di duplicare i fotogrammi simili sottraendo al calcolatore i tempi di calcolo necessari per renderizzarli tutti in maniera ripetitiva. Se vi si ponesse un caso simile, ecco come comportarvi. Il programma per cui questo accorgimento risulta valido è naturalmente il grande *Imagine* della Impulse. Renderizzate normalmente per intero il primo fotogramma della sequenza che prevede la serie di fotogrammi identici. Poniamo che la sequenza sia formata da 10 fotogrammi tutti uguali: renderizzate allora solo il frame 1. Dopodiché per ciascun fotogramma successivo e identico (nel nostro caso per i 9 frame rimanenti) dovete adottare un semplice espediente pratico. Impartite l'ordine di rendering per il fotogramma 2, il programma caricherà oggetti, texture e brush (fase di loading objects indicata con messaggio sulla Title Bar), dopodiché inizierà una fase preliminare in cui *Imagine* ordinerà i vari componenti (sulla barra in alto apparirà la dicitura "initializing..."). Dopo un tempo più o meno lungo (dipende dalla complessità della scena), il programma passerà a scrivere sempre sulla Title Bar: "Generating..." seguito dalla percentuale di completamento. Selezionate Cancel nel requester centrale e il programma vi risponderà, interrompendo i calcoli, con un requester del tipo: "Delete still file(s)?" Selezionate No e seguirà una fase di "Cleanup", sempre palesata da un messaggio sulla Title Bar. Per finire, *Imagine* terminerà il tutto e porrà un asterisco sotto al fotogramma calcolato (per indicare che è stato calcolato, ma in realtà è incompleto in quanto interrotto prematuramente, prima cioè che la percentuale di calcolo raggiungesse il 100%). Dal momento che il fotogramma è stato calcolato solo in parte, se vi recherete con un programma di servizio e gestione file (*Diskmaster* o *Directory Opus*) all'interno del progetto (nomefile seguito dal suffisso .imp), e poi dentro alla directory automaticamente creata da *Imagine* col nome del sottoprogetto seguito dal suffisso .pix,

troverete nel nostro caso il file renderizzato chiamato pic.0001 e il file interrotto pic.0002 più piccolo, perché naturalmente incompleto. Allora copiate pic.0001 in ram:, rinominatelo in pic.0002 e ricopiatelo nella stessa directory al posto di quello incompleto.

Semplice, no? Ripetete ora il procedimento per ciascun fotogramma identico e il gioco è fatto, *Imagine* procederà a riassemblare l'animazione (i fotogrammi identici occuperanno pochissimo spazio nella compressione delta) e voi vi sarete risparmiati un mucchio di tempo. Per quanto riguarda la mia animazione, il risparmio totale di tempo è stato superiore alle 300 ore!

Arrivano le lenti anche con Real 3D

L'affezionato, esperto e attentissimo lettore Saverio Spinelli di Reggio Calabria ci ha scritto per una precisazione importante riguardo alla rubrica *Techno 3D* del numero 1/93 dedicata alla modellazione delle lenti. Sebbene i parametri e il funzionamento fisico fossero spiccatamente dedicati agli utenti di *Imagine*, affermavo nell'articolo che i principi di modellazione contenuti potevano essere estesi a qualsiasi pacchetto di ray tracing ed elencandoli facevo anche notare che questo non era possibile con *Real 3D* della Activa. La cosa però non era completamente vera (era necessario assumere dei particolari accorgimenti e comportamenti in modellazione) e il lettore ha pensato bene di puntualizzare, allegando come prova anche un'immagine calcolata con *Real 3D* versione 1.4.2 che potete vedere nella pagina seguente.

Ma lasciamo la parola al lettore: "[...] Vi invio un'immagine con lenti realizzata con *Real 3D* dalla quale si evince come anche con questo splendido pacchetto sia possibile riprodurre leggi fisiche che regolano l'ottica (l'unica cosa che non si può ottenere è la separazione della luce nei colori dell'iride quando questa passa attraverso un prisma in quanto nel calcolo della rifrazione non si tiene conto della diversa lunghezza d'onda delle radiazioni luminose colorate). Con *Real 3D* si ottengono quindi lenti otticamente perfette. Per modellare le mie lenti ho utilizzato un sistema molto semplice: ossia ho eseguito un'operazione booleana di "AND" tra due sfere, assegnando poi le priorità (attributi) tipiche del vetro (colore bianco, elevate riflessioni e trasparenze, indi-

ce di rifrazione pari a 1.6). L'immagine è in risoluzione 640 x 512 e ha richiesto 30 minuti di calcolo".

Complimenti al nostro esperto lettore, che, oltre all'esatta puntualizzazione, apre una discussione che forse merita qualche contributo da parte di altri lettori poco o molto esperti. È possibile con un qualsiasi programma di ray tracing ottenere la separazione dei colori dell'iride per mezzo di un prisma (naturalmente senza artifici di simulazione come brush proiettati con i colori dell'iride...)? Tratterò l'argomento prossimamente e pubblicherò gli interventi più interessanti eventualmente giunti. Spremete dunque le meningi (e consultate i libri di fisica ottica...). Oltre a quanto trattato ci sono

dettagli numerosi e minuti. Mi è successo durante la modellazione della sagoma dentellata di un francobollo che il programma si bloccasse durante la fase di salvataggio in formato Image (mentre i brush vengono acquisiti senza problema). Allora ho prelevato il brush disegnato con *D-Paint* e l'ho importato all'interno di *Art Department Pro*, dopodiché ho riscalato le dimensioni ingrandendole del doppio e stando attendo in *Execute* a non selezionare il dithering.

Salvato il disegno così approntato, *Pixel Pro* non ha avuto più problemi a eseguire una conversione ottimale. L'accorgimento non basta a risolvere tutti i problemi, ma può rivelarsi utile adottarlo in molti casi.

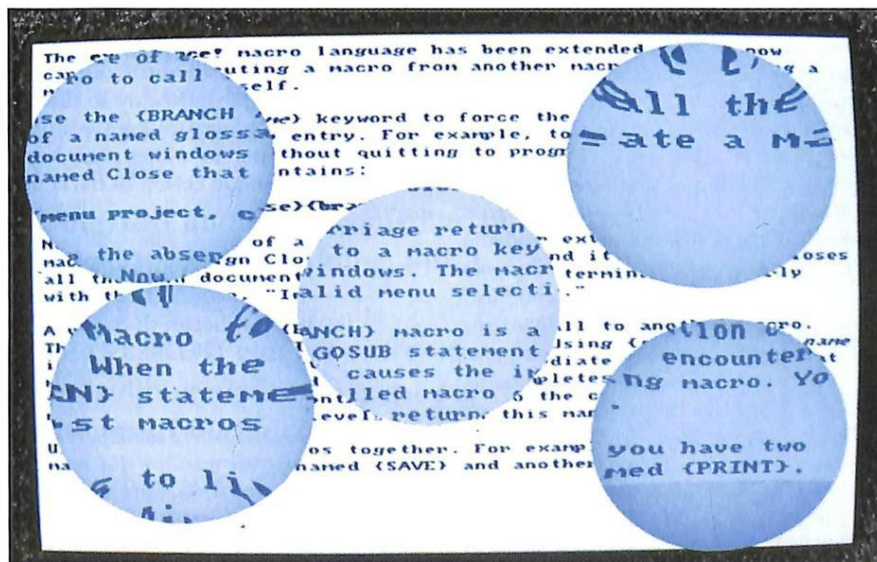
presente sotto il menu Object. Automaticamente, le texture dell'asse saranno estese all'oggetto figlio. Se volete ancora altre quattro texture a disposizione, ripetete la procedura con un secondo asse con definite altre diverse texture, e così via. Attenzione però al tipo di texture e alla loro gestione. Queste, infatti, sono gestite sotto forma di layer (strati) e quindi le varie possibilità di sovrapposizione dipendono dal tipo di texture selezionata e da quanto ognuna di esse faccia trasparire gli strati sottostanti. Se pertanto saturate i vari layer non vedrete la sovrapposizione di tutte le texture.

Tutte le texture che desiderate! (2)

Memorizzare le poche tessiture fornite con *Imagine* su hard disk non comporta che l'occupazione di 30K scarsi di spazio. Se però insieme a questi si affiancano anche le texture contenute nella libreria *Essence* di Steve Worley (si veda in proposito la recensione presente in questo stesso numero) i Kbyte totali possono salire a oltre 800! Pochi sanno, però, che è possibile comprimere questi file con l'ausilio di *PowerPacker* (o un altro programma che contempli la possibilità di scompattazione automatica in fase di richiamo del file, quindi non potrete utilizzare compattatori quali i vari *lharc*, *pkzip*...), mantenendo pressoché inalterata la loro funzionalità!

Messaggi nascosti

Nonostante la documentazione non ne faccia cenno, se provate a leggere con qualsiasi editor o word processor i file che compongono le singole texture di *Essence*, potrete rilevare una gradita sorpresa. All'interno di ogni file vi è una breve spiegazione sull'effetto della texture e persino sulle specifiche da impostare nei vari parametri. Probabilmente, Steve Worley, autore della collection, ha ricevuto dalla Impulse direttive sulle prossime implementazioni delle texture algoritmiche presentate con *Imagine* release 3. Vale a dire la presenza di un help in linea o quantunque di uno spazio di poche righe di testo all'interno del riquadro di specifica che descriva, una volta caricata la texture, le modalità d'impostazione e l'azione.



Un esempio di lenti con Real 3D realizzato dal nostro lettore Saverio Spinelli

ancora moltissimi comportamenti fisici della luce che è possibile riprodurre. Anche in base alle vostre esperienze scolastiche di Fisica (mi riferisco per lo più a studenti di licei scientifici e tecnici oppure a studenti universitari) non avete mai provato a utilizzare un programma di ray tracing per compiere interessanti esperimenti sul comportamento della luce?

Brush minuti con Pixel 3D

Sembra che *Pixel 3D* (Axiom), sia in versione 2 che in versione Pro, abbia qualche problema quando i contorni di un brush da trasformare in oggetto 3D risultano troppo piccoli e composti da

Tutte le texture che desiderate! (1)

C'è un'interessantissima finezza conosciuta da pochissimi che consente di aggirare il limite delle quattro texture algoritmiche massime consentite da *Imagine* per ciascun oggetto. Senza scomodare i sottogruppi, o peggio spezzettare l'oggetto in più parti, è possibile aggiungere un numero a piacere di texture con un semplice quanto comodo accorgimento. Una volta definito l'oggetto con le solite quattro texture, aggiungete un semplice asse (Add Axis). A quest'ultimo conferite le altre texture selezionate e abbiate cura di selezionare "Apply to Child Objects". Ora selezionate prima l'asse in questione, e poi l'altro oggetto, e raggruppate il tutto col comando Group

POSTA 3D

Il filo diretto tra i lettori e il mondo del 3D: racconti, domande e risposte...

Questa rubrica ospita i quesiti e le impressioni dei lettori concernenti gli aspetti più vari della grafica 3D. Indirizzate la vostra corrispondenza a: *Commodore Gazette*, Rubrica Amiga 3D, Via Monte Napoleone 9, 20121 Milano. Oppure potete rivolgervi direttamente alla casella presente nella BBS New Horizons di Roma dedicata a *Imagine*, lasciando un messaggio all'attenzione di Antonio De Lorenzo (BBS 06/8862660 - 88640190).

UNA GRANDE PASSIONE MI BRUCIA IL COR...

Sono un ragazzo di 19 anni e adoro la grafica 3D e soprattutto *Imagine 2.0* della Impulse. Non sono una persona molto eloquente... l'unica cosa che vorrei chiedere è di dare un'occhiata alle mie immagini (tutte in HAM) accluse sul dischetto allegato e di giudicare se ho qualche possibilità di entrare nel mondo della grafica; questo sarebbe il mio sogno visto che del mio futuro diploma di ragioniere non me ne importa molto. Il mio sogno è quello di vivere davanti a una stazione grafica e basta, non m'importa di guadagnare molto, i soldi non sono la mia passione, mi basta il necessario per sopravvivere.

Tutti gli oggetti presenti nelle immagini, tranne il maggiolino Volkswagen sono miei originali, compresa l'idea e la scenografia.

Tiziano Nocentini
Agliana (Pistoia)

Devo confessare che la sua lettera mi ha sconvolto non poco. Comprendo la sua passione per la grafica 3D, ma non a tal punto da farla propendere per un abbandono sulla scelta d'un impiego legato alla scuola frequentata. Poi quel "... sogno di vivere davanti a una stazione grafica e basta..." mi sembra un tantino esagerato (sembra la nascita di una di quelle creature in parte umane e in parte di silicio tanto care alla letteratura fantascientifica cyberpunk). La vita non è fatta di soli calcolatori e grafica 3D (c'è pure quella 2D! Sto scherzando naturalmente...) e spero che lei intendesse dire che professionalmente le piacerebbe trascorrere la maggior parte del suo tempo a

fare grafica 3D.

La passione per la grafica 3D nasce per forze di cose come un hobby e richiede del tempo per essere portata a livelli professionali, non so quanto convenga affidarsi dall'inizio a una passione seppure grande come quella che lei mostra sicuramente di nutrire. Il mio consiglio è di coltivare la sua grande passione come hobby e farla crescere insieme ad altri interessi, se poi questa richiederà più tempo e i risultati la soddisferanno a pieno, potrà decidere di farla divenire una vera e propria attività lavorativa, dedicandovisi esclusivamente.

Un giudizio sulle sue immagini. Iniziamo dagli oggetti. Noto con piacere che segue con attenzione *Techno 3D*, mentre la maggior parte di essi sono di provenienza esterna (non solo il maggiolino, ma anche la bottiglia di profumo e l'oggetto spiraliforme in forma di toro). Il lavoro mostra alcune ingenuità di accostamenti, soprattutto nella realizzazione delle scenografie (non se ne faccia comunque un cruccio le immagini di tutti gli appassionati alle prime armi sono molto al di sotto della qualità generale riscontrata nelle sue). Alcuni oggetti sono comunque molto ben realizzati, in particolare è degna di nota la scatola di mentine Tabu per il realismo e le note con gli strumenti musicali. Noto anche una ricerca dei materiali e delle texture che sebbene ancora in fase embrionale contiene senz'altro le premesse per riuscire ottimamente nel campo.

Inoltre, le consiglio di evitare la ricerca a tutti i costi del realismo con lucentezze e riflessioni. Parta sempre da un'idea, cerchi d'includervi un messaggio e poi faccia di tutto per realizzarla, si accorgerà come questa migliorerà da sola durante la realizzazione. L'immagine deve comunicare qualcosa, altrimenti resta una sterile sperimentazione.

Concludendo, lei possiede tutte le carte in regola per riuscire ottimamente nel campo ed è pronto ad abbandonare le prime due fasi (oggetti iper-riflettenti, mostra di essi in quanto tali, effetti ricercati) per entrare nella terza, quella più interessante e stimolante basata sulla comunicazione.

Buona fortuna. E perché no, ci riscriva fra un po' di tempo, inviandoci le sue nuove realizzazioni.

UN LUNGOMETRAGGIO CYBERPUNK

Vi scrivo a nome di un gruppo di giovani cineasti, dopo aver letto l'articolo riguardante il Bit Movie. Infatti, stiamo cercando di realizzare un lungometraggio con una sceneggiatura di tipo cyberpunk e una delle cose che abbisognano al filmato sono proprio le animazioni al computer.

A questo proposito ho due richieste da farvi, sperando vivamente che vengano accolte. La prima riguarda le informazioni (indirizzo, numero di telefono) di coloro che hanno vinto (o soltanto partecipato o che sono disposti a collaborare con noi anche se non hanno partecipato) il Bit Movie, sperando che se contattati accettino un'eventuale collaborazione con noi. La seconda invece riguarda la fornitura d'indirizzi di ditte disposte a fornire del materiale hardware in cambio di pubblicità: il nostro budget è infatti molto limitato perciò è ben accolta la presenza di chiunque cerchi di darci una mano.

Vittorio Cristiano
Via Locanda, 22
71044 Margherita di Savoia (FG)
(Tel. 0883/654441)

Sappiamo da altre lettere simili pervenute che iniziative come quella della realizzazione di lungometraggi, raccolte d'immagini e/o animazioni... stanno divenendo sempre più diffuse da parte di molti nostri lettori intraprendenti.

Non possiamo fornirvi le informazioni riguardo i partecipanti del Bit Movie, perché si tratta d'informazioni non in nostro possesso e poiché comunque gli indirizzi non possono essere divulgati se non dietro autorizzazione degli interessati. Può rivolgersi al direttore artistico del Bit Movie Carlo Mainardi all'indirizzo che potrà trovare nella pagina del Bit Movie Art.

Ogni autore risponde delle proprie opere e stabilisce le condizioni di cessione. Stesso discorso riguarda l'indirizzo di fornitori di materiale software e hardware.

In via eccezionale, pubblichiamo il vostro recapito e invitiamo tutti i lettori e le ditte interessate a contattarvi per discuterne personalmente.

Per quanto ci riguarda, potremo dare pubblicità alla vostra iniziativa qualora ci terrete informati sullo stato dei lavori e ci farete conoscere il livello finale della vostra realizzazione.



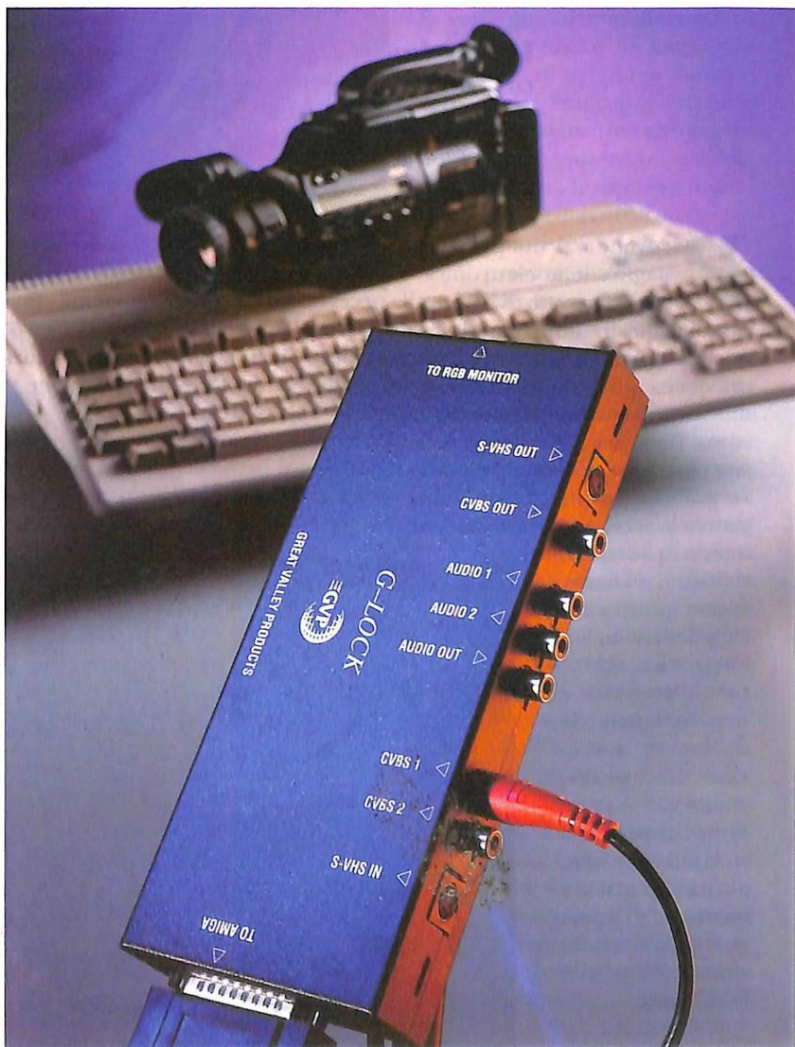
Presenta

G-LOCK: il nuovo Genlock Professionale

Con **G-LOCK**, puoi creare Video ed effetti multimediali, utilizzando simultaneamente Video, Grafica, Audio AMIGA semplicemente con un «click»

Caratteristiche:

- Pannello di controllo a “bottoni”, estremamente intuitivo. Interfaccia Arexx & Cli.
- Software switchabile tra: 2 ingressi Videocomposito oppure 1 ingresso Y/C Super VHS.
- Tempo reale, Video processore (Proc Amp) controllato via software con gestione totale del segnale video. Compatibilità totale con digit audio GVP DSS.
- 2 ingressi audio con controllo, in tempo reale, del Volume, alti, bassi Mix & Mute.
- Splitter RGB con software di controllo compatibile con i più conosciuti digitalizzatori.
- Complete operazioni di trascode con uscite Videocomposito, Y/c e RGB/YUV.
- Compatibilità totale con AMIGA VISION, SCALA, SCALA MM.



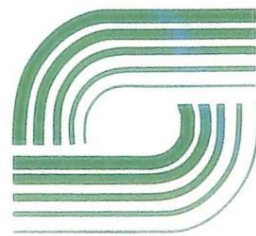
TUTTO QUESTO, E MOLTO PIÙ, AD UN PREZZO INCREDIBILE!
DAL TUO RIVENDITORE DI FIDUCIA O DAL GVP Point PIÙ VICINO.



è distribuito da:

RS ricerca e sviluppo s.r.l.

VIA B. BUOZZI, 6 - 40057 CADRIANO DI GRANAROLO (BO)
TEL. 051/765563 - FAX 051/765568 - BBS. 051/765553



PROVE HARDWARE/SOFTWARE

ALFASCAN: SCANNER A TONALITÀ DI GRIGI

Un package completo per acquisire testi e immagini che comprende anche un interessante programma di OCR. Il tutto a un costo accessibile

di Marco Dufour

Provando alcuni programmi d'im-paginazione elettronica, mi sono spesso trovato di fronte alla necessità di acquisire immagini da inserire nei miei stampati. La soluzione finora seguita era quella di digitalizzare le fotografie da una sorgente video tramite digitalizzatori come DigiView, VidiAmiga e VideoMaster. Anche se i risultati potevano sembrare più che buoni, la scomodità di uno stativo per telecamera e l'attenzione che questa procedura richiede, mi hanno spesso spinto a cercare delle soluzioni alternative, non ultima l'acquisto di uno scanner. La differenza sostanziale del sistema scanner è la precisione nei particolari, la maggior semplicità di posizionamento dell'immagine e l'alta risoluzione che è possibile ottenere.

Nell'utilizzo di AlfaScan Plus, l'ultimo nato nella famiglia degli hand scanner, si riscontrano tutte le doti di un ottimo prodotto: maneggevolezza, praticità d'uso, alta definizione e buon software di supporto.

Ma cos'è uno scanner e più precisamente un hand scanner? Per chi proprio con i computer non abbia molta familiarità, ricordiamo che uno scanner è una specie di macchina fotocopiatrice dove le immagini, anziché essere copiate su carta, vengono acquisite dal computer. In parole povere, un fascio di luce colpisce la carta, viene illuminata una linea e l'apparecchio ricompone l'immagine a seconda

dell'intensità della luce riflessa. La densità di queste linee parallele determina la qualità dell'immagine finale: si parla quindi di punti per pollice o dpi (dots per inch) per indicare la risoluzione massima sia in fase di acquisizione, sia in fase di stampa.

Quando si parla di scanner in generale, si pensa sempre ai modelli a piano fisso (flatbed) in formato A4. Questo particolare tipo si rivela molto utile se si ha bisogno

ha la necessità di scansire immagini di grandi dimensioni.

Mettiamo da parte le esigenze super-professionali e analizziamo cos'è possibile fare spendendo meno della metà di uno scanner flatbed.

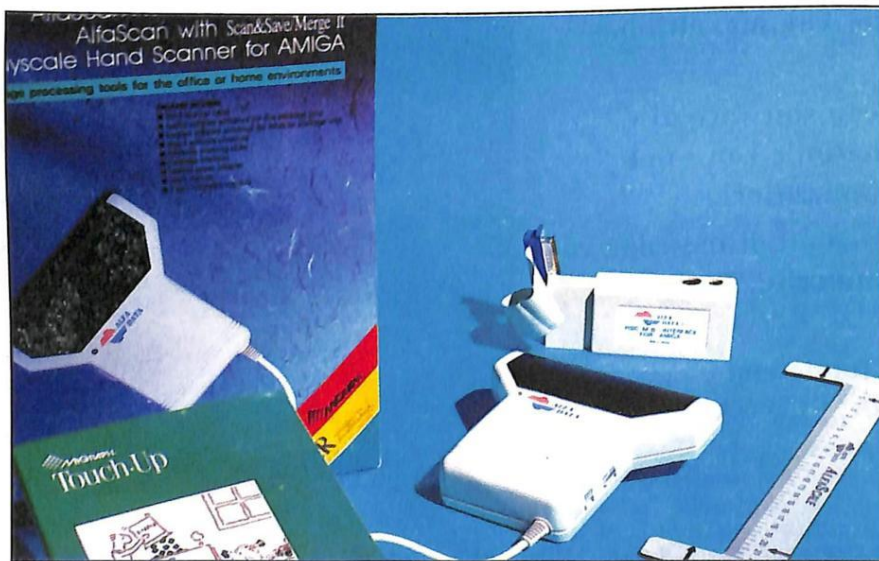
La dotazione

AlfaScan Plus è composto da due corpi distinti tra loro. Uno è propriamente il "motore" dell'operazione mentre l'altro è l'interfaccia per il collegamento all'Amiga. Quello che ho chiamato motore è lo scanner vero e proprio. Chi avesse un po' di familiarità col mondo PC, può notare senza difficoltà l'estrema somiglianza di AlfaScan con altri apparecchi di questo genere esistenti sul mercato. In effetti, il "motore" è lo stesso o molto simile a quello fornito da

altri modelli di aziende diverse: la differenza tra i vari prodotti è data dall'interfacciamento al computer e dal programma di gestione dello stesso scanner.

AlfaScan si collega a qualsiasi modello di Amiga tramite la porta parallela, che permette un più veloce trasferimento dei dati tra le due apparecchiature. Viene fornito anche un trasformatore da 12 volt e un righello-guida per la scansione d'immagini di grande formato.

Per l'acquisizione delle immagini sono necessari almeno 2,5 MB di RAM; come



di acquisire immagini di grandi dimensioni ma ha, almeno per ora, un prezzo ancora elevato. AlfaScan Plus rientra invece nella schiera degli scanner manuali (hand scanner), dove il movimento del fascio di luce è accompagnato da un movimento della mano. I pregi e i difetti dei due diversi sistemi sono facili da individuare: il sistema flatbed è decisamente più pratico, ma più costoso, il sistema hand scanner: se da un lato è meno ingombrante e più economico, dall'altro risulta più difficile da usare se si

ormai con quasi tutti i programmi, più memoria si ha, più sono le possibilità. Le prove sono state eseguite su un Amiga 4000. Alcuni test sono stati effettuati anche su un Amiga 1200 con hard disk, raggiungendo in entrambi i casi buoni risultati.

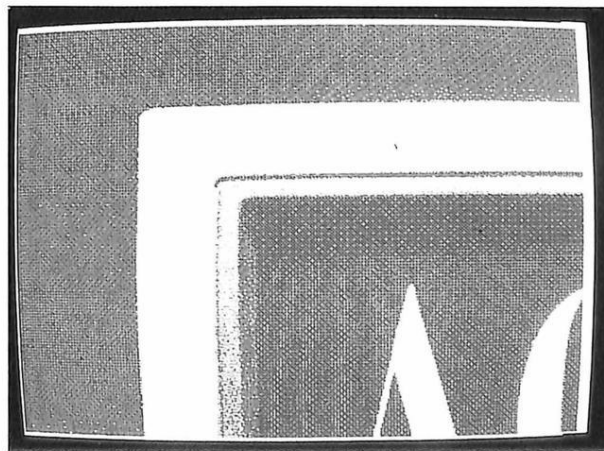
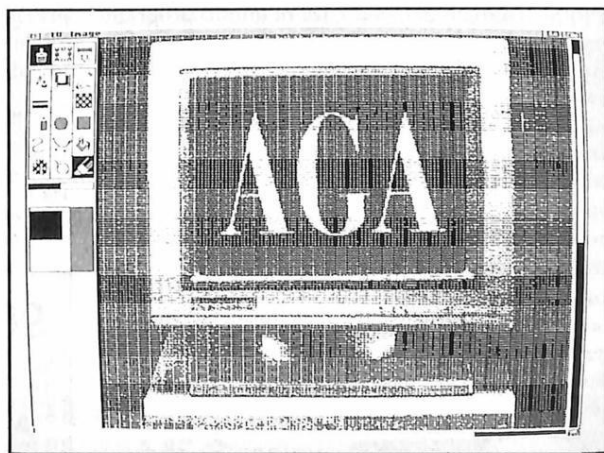
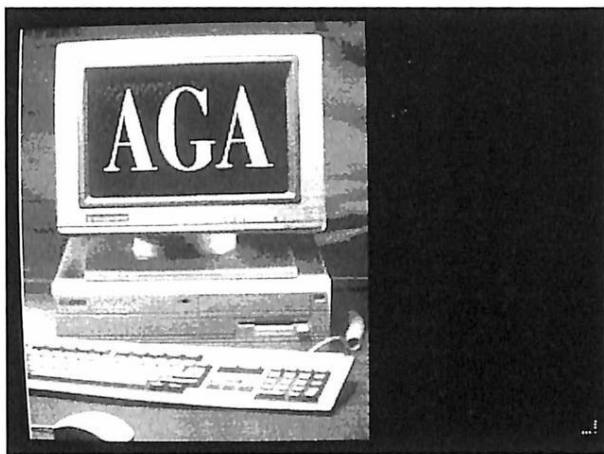
Il programma Touch-Up

Sempre compresi nella confezione troviamo tre importantissimi programmi: *Touch-Up*, per la scansione e il ritocco delle immagini, *Merge-It*, per comporre immagini tramite la scansione in più passate, e OCR, il primo programma per Amiga in grado di eseguire il riconoscimento dei caratteri.

La versione di *Touch-Up* è la 2.50. Questo programma della Migraph, sebbene forse un po' datato (risale al 1991), si comporta magnificamente con il sistema operativo 3.0 e non ha problemi neanche con le altre revisioni. *Touch-Up* a sua volta è suddiviso in tre menu principali: quello con i tool di disegno, quello con gli operatori di image processing (rotate, slant, bold...) e quello dedicato all'acquisizione delle immagini. Nelle tre diverse sessioni di lavoro il puntatore del mouse cambia di aspetto, facilitando molto il riconoscimento da parte dell'utente. Si nota invece una certa "atarizzazione" dell'interfaccia, chiaro segno che le prime versioni di questo programma funzionavano sui modelli ST dell'Atari.

Il primo passo per scansire un'immagine è decidere la risoluzione finale su computer: l'immagine acquisita verrà mostrata da un retino in bianco e nero, tanto più preciso quanto più alta sarà la risoluzione scelta. Si può quindi decidere se utilizzare 100, 200, 300 o 400 dpi (punti per pollice), tenendo presente che tanto più alta sarà la risoluzione, maggiore sarà la richiesta di memoria. Si porterà quindi lo scanner sulla porzione desiderata e, dopo aver dato il comando sul computer, basterà schiacciare il pulsante sull'apparecchio e cominciare lo scorrimento sul foglio. A seconda dell'immagine sorgente, se cioè si tratta di una foto o di una pagina di testo, si sceglierà il tipo di dithering da adottare tramite un'interruttore posto sulla sini-

stra dello scanner stesso. Sulla destra, oltre al comando di brightness, per la regolazione della luminosità, troviamo l'interruttore per decidere la risoluzione da utilizzare espressa in punti per pollice.



Sopra: un'immagine convertita in toni di grigio impiegando l'algoritmo di Touch-Up. Al centro: l'interfaccia di lavoro di Touch-Up. Sotto: un dettaglio dell'immagine in bianco e nero

Nel giro di pochi secondi, se i collegamenti sono fatti correttamente, si ha sullo schermo dell'Amiga l'immagine. A dire la verità, quella che avremo all'interno di una finestra di *Intuition* sarà solo una rappresentazione in alta risoluzione di

quello che lo scanner ha trasmesso al computer. Quando sulla scatola dell'apparecchio ho letto "256 tonalità di grigio", mi sono entusiasmato all'idea di poter acquisire con altissima precisione qualsiasi immagine. In realtà, tutto avviene in bianco e nero. Sarà poi il programma a trasformare l'immagine in tonalità di grigio tenendo conto della diversa concentrazione dei punti a seconda dell'intensità della luce. È vero quindi che il risultato finale può essere un'immagine in tonalità di grigi, ma è anche vero che le immagini ottenute con questo algoritmo risulteranno fino a un quarto più piccole dell'originale. Il sistema utilizzato da *Touch-Up* non è comunque tra i migliori: avendo la possibilità di provare questo comando anche con *Image F/X* della GVP, ho riscontrato un risultato quasi perfetto con quest'ultimo, mentre in paragone un po' scadente con il programma di gestione dello scanner. Bisogna anche dire che il risultato ottenuto non è in grado di far mal giudicare il software *Touch-Up*: se il nostro utilizzo è prettamente destinato all'impaginazione elettronica, la qualità dell'immagine in toni di grigio può anche non interessarci: l'immagine in bianco e nero è l'ideale per la stampa e la sua retinatura risulta perfetta per questi usi. Se invece pensate di acquisire immagini per fare fotoretocco (utilizzando programmi di grafica), il mio consiglio è di affiancarlo a *Image F/X*, il che vi permetterà di ottenere risultati davvero interessanti. Una volta acquisita l'immagine, è possibile ritoccarla sia con tool di disegno classici (pennello, fill, gomma...), sia ruotandola o operando funzioni di slant (inclinazione rispetto a un asse). A questo punto bisogna dire che per quanto riguarda questa sessione di lavoro il programma si comporta in maniera eccellente, senza mai, o quasi, far sentire la mancanza di programmi come *Deluxe Paint*.

La superficie di scansione di AlfaScan Plus è di dimensioni ridotte (10,5 centimetri): se si vogliono scansire immagini di dimensioni superiori, l'unico sistema è quello di gestire l'operazione in due o più passate, tramite l'aiuto del righello guida e del programma *Merge-It*. L'interfaccia di questo programma è molto semplice: con un

comando si scelgono le due immagini da incollare e con un altro si sceglie il nome dell'immagine finale. Il programma si occupa poi di tutto il resto.

L'operazione in sé non risulta complicatissima, ovvio però che si comincia a invidiare la gestione del formato A4, caratteristica di punta degli scanner flatbed (con questo scanner un foglio A4 si scansisce in due passate). Con un po' di pratica, si supera anche questo ostacolo, risparmiando notevoli quantità di denaro a fronte di un processo più laborioso.

Ed ora vediamo l'OCR

Con *Optical Character Recognition* s'intende un programma in grado di leggere, partendo da un'immagine bitmap, un testo in formato ASCII. Mettiamo per esempio di dover trascrivere un articolo di giornale per poi correggerlo o reinserirlo all'interno di un'altra pubblicazione: fino a poco tempo fa l'unica soluzione risultava utilizzare un programma di videoscrittura e con un po' di pazienza trascrivere l'articolo a mano. Grazie invece a questo programma, o simili, è possibile far svolgere questa operazione in maniera quasi automatica dal computer. Partendo da una scansione, il computer analizzerà lettera per lettera il nostro testo, cercando la somiglianza dei caratteri con un suo alfabeto interno impostato in fase di lettura. Ai più forse non è sfuggito che un procedimento del genere è da considerarsi Intelligenza Artificiale: il computer cioè, partendo da un set d'istruzioni predefinito dal programmatore, è in grado di aggiornare il suo vocabolario e di svolgere un'operazione in modo molto simile a quella umana.

Umanamente parlando, almeno apparentemente, la lettura di un testo non è assolutamente un processo complicato e lento; il computer, invece, trova molte difficoltà a riconoscere i caratteri e non ha l'intelligenza sufficiente per uscire dalle informazioni impostate: se un carattere è scansito in maniera non perfetta, gli risulterà ancora più difficile il riconoscimento. Per accelerare il processo di apprendimento, c'è la possibilità di creare veri e propri dizionari di parole che il calcolatore di volta in volta consulerà. Se per esempio la parola "Computer" venisse male interpretata dal programma fino a leggere "Compater", il dizionario sarà in grado di avvertire l'OCR che il vocabolo è inesistente, che la forma corretta è "Computer", e chiederebbe la sostituzione o l'inglobazione del nuovo vocabolo nel dizionario.

A parole sembra molto semplice. Quando poi si va a lavorare sul serio, si scopre che il corpo 12 dei caratteri

(utilizzato normalmente da diverse pubblicazioni) risulta troppo piccolo per un riconoscimento automatico, anche se scansito al massimo della risoluzione. Lettere come per esempio "sc" tendono a impastarsi tra loro, rendendo molto difficile il riconoscimento. Dopo vari tentativi, ho trovato la soluzione d'ingrandire il testo desiderato tramite fotocopia e scansarlo al massimo della risoluzione. Questo procedimento è forse il più efficace e permette un lavoro più automatico.

L'alternativa è scansare il testo normale e passare molto tempo a correggere il testo acquisito, facendo uso di un qualsiasi word processor. Ho calcolato che anche se il tempo non si avvicina a quello impiegato per trascrivere il testo a mano, il processo risulta un po' snervante: in alcuni casi bisogna addirittura lasciar perdere questa possibilità e lavorare direttamente sulla tastiera.

Sono rimasto comunque molto impressionato dalla potenza di questo programma che, nonostante sia ancora alla versione 1.11, si comporta in maniera abbastanza buona. Non mi sento di dire che è perfetto, che è talmente automatizzato che non bisogna assolutamente fare nulla. Bisogna fare un po' di pratica, scoprire qualche trucco per fargli leggere al meglio il testo e... il lavoro risulta comunque dimezzato. Se il vostro obiettivo d'acquisto era un OCR, AlfaScan è senza dubbio una soluzione interessante. Bisogna poi considerare che anche i programmi di OCR per gli altri computer soffrono degli stessi problemi di questo prodotto della Migraph.

Conclusioni

AlfaScan è un buon prodotto e i programmi di supporto sono abbastanza potenti da competere con la concorrenza. La nitidezza delle immagini acquisite ne fa una scelta interessante. In alcuni ambiti di utilizzo AlfaScan risulta comunque particolarmente competitivo. Sicuramente, posso consigliarlo a chi si interessa di DeskTop Publishing. Considerando che una nuova versione di *Touch-Up* è già stata presentata all'estero, e che i risultati in stampa sono ottimi, diventa uno strumento indispensabile. Un altro possibile utilizzo può essere l'acquisizione di documenti da spedire via fax: grazie ad alcuni modem messi in commercio, è ormai possibile spedire facsimili tramite il computer, ottenendo qualità di riproduzione praticamente perfette. Attualmente, programmi come *GP Fax* (a corredo dei modem SupraFax) permettono di trasformare documenti impaginati con qualsiasi programma in pagine fax da spedire via modem. Quello che invece non si può

fare, è inserire documenti esterni: ecco quindi che per mettere per esempio una firma all'interno di un documento la soluzione più pratica risulta quella di scansarla. Ho personalmente provato queste possibilità, riscontrando una grande praticità di lavoro.

Va anche citato il fatto che il programma di gestione consente di salvare i file generati nei formati EPS, IFF, IMG, MAC, PCX, P13 E TIFF. La manualistica acclusa (in lingua inglese) è piuttosto chiara ben realizzata ed è presente anche una cartolina che parla di una garanzia di due anni.

AlfaScan Plus viene venduto a 450 mila lire (Iva compresa) completo di programmi, trasformatore e interfaccia; come ho già tenuto a dire precedentemente, non è un prodotto per tutti (l'utenza più professionale preferirà infatti scanner flatbed a colori come l'Epson GT-6500, si veda in proposito il numero 2/93), ma visto il prezzo, non posso che consigliarlo a tutti gli interessati all'impaginazione elettronica sull'Amiga.

Il prodotto recensito è disponibile presso:

Data Office

(AlfaScan: L. 450.000, Iva compresa)
Via Roma, 517
80040 S. Sebastiano al Vesuvio (NA)
(Tel. 081/5743260)

SCHEDA CRITICA

Prodotto:

ALFASCAN PLUS

VOTO:

(In decimi)

8,0

Funzionalità:	★	★	★	★	★	
Conferma aspettative:	★	★	★	★	★	
Design:	★	★	★	★	★	
Affidabilità:	★	★	★	★	★	
Tecnologia:	★	★	★	★	★	
Documentazione:	★	★	★	★	★	
Prezzo/prestazioni:	★	★	★	★	★	

Che cos'è: un sistema integrato hardware e software per il trasferimento d'immagini dal supporto cartaceo al computer. Lavora in bianco e nero, ma il suo software può convertire le immagini in 256 tonalità di grigio. La superficie di scansione è di 10,5 cm.

Cosa ci è piaciuto: la praticità di questo sistema lo rende una soluzione attraente anche per chi dovesse usarlo solo saltuariamente. La qualità dei risultati e la semplicità dei programmi di supporto.

Cosa non va: sebbene abbia raramente dato problemi di funzionamento, sarebbe interessante poter acquistare il prodotto con la nuova versione dei programmi, visto che sono già commercializzati. I problemi riscontrati, come la qualità della conversione in tonalità di grigio, dipendono esclusivamente dal software.

VIDEO BLASTER

Interfaccia multimediale per PC



Acquisisce immagini video composte da tre sorgenti diverse, supporta sistemi NTSC e PAL. Acquisisce nei formati PCX, TIFF, BMP, MAMP, GIF e FAXGA. Risoluzione 640 x 480. Mixer audio, amplificatore di suono stereo.

Compatibile PC-A1 e superiori. Slot 1x bit. MS-DOS 3.1 e superiori.
Monitor VGA o Multisync 50-70 Hz.
£ 599.000

AUDIO BLASTER

Scheda multimediale. Software professionale di composizione e performance musicale per Windows, Connessione MIDI, Set di giochi PC, Sintetizzatore, Mixer.
Per tutti i PC compreso XT.
£ 199.000

SOUND BLASTER
PRO 2 MULTIMEDIA
L. 430.000

Schede VGA per PC con 2 Mb
NCR RAM DAX XGA
£ 330.000

NOVITÀ COMMODORE PC
MS DOS 5.0 con licenza
MONITOR SUPER VGA 14" Color

486 SLC 25MHZ CPU • 2 MB RAM • CACHE INTERNA CPU • HARD
DISK 120 MB 11 Ms • TASTIERA ESTESA 102 TASTI • 2 Seriali, 1 Seriele PS 2
1 Parallela • Schede VGA Super 16.8 Milioni di colori

IN OMAGGIO AI PRIMI 50 ACQUIRENTI
UNA STAMPANTE 80 COLONNE BIDIREZIONALE
GRAFICA 200 CPS MODULO CONTINUO E CARTA LETTERA MPS 1230

£. 1.800.000
* IVA



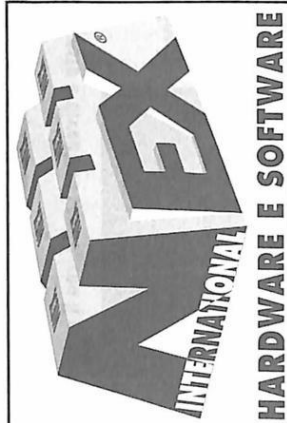
Vidi PC 12 è un digitalizzatore video a 12 bit per PC. Visualizza immagini a 4096 colori con una risoluzione massima di 1024 x 768 sulle schede VGA supportate. Il tempo di digitalizzazione è meno di 1 secondo per le immagini a colori e in tempo reale a 16 livelli di grigio da segnali video in movimento. **£. 499.000**

Vidi PC 24 è un digitalizzatore video a 24 bit per PC. Visualizza immagini a 4096 colori con una risoluzione massima di 1024 x 768 sulle schede VGA supportate. Il tempo di digitalizzazione è meno di 1 secondo per le immagini a colori e in tempo reale a 256 livelli di grigio da segnali video in movimento. **£. 749.000**

Media Pro Plus digitalizza in tempo reale i segnali video PAL e S-VHS. **£. 899.000**

Tutte le versioni sono fornite di software di gestione compatibile co Windows e DOS

SCANNER MANUALE
PROLAB 400 DPL, 256
TONALITÀ + OCR L. 399.000

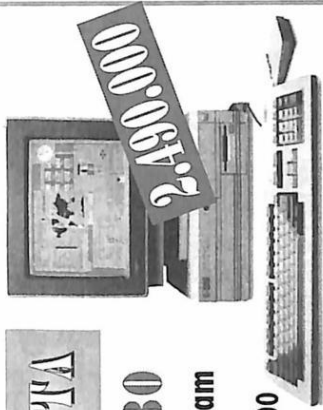


DI ANTONIO CIAMPITTI
NEGOZIO DI VENDITA AL PUBBLICO **02/93505280**
VIA BUGATTI, 13 - 20017 RHO (MI)
PER LE ORDINAZIONI **02/93505942**
POTETE TELEFONARE ALLO
OPPURE MANDARE
UN FAX ALLO **02/93505219**
SPEDIZIONI IN TUTTA ITALIA PER POSTA O CORRIERE

NOVITÀ ASSOLUTA

AMIGA 4000/30/380
68030 con 80 Mb HD - 4 Mb Ram

Scheda di interfaccia SCSI per Amiga 4000
L. 80.000



AMIGA 4000

il potentissimo computer Commodore
con prestazioni da workstation.

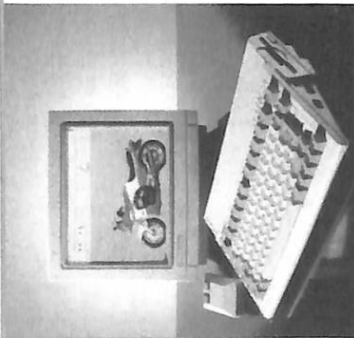
CPU Motorola 68040
Clock 33 MHz, 256.000 colori
AmigaDOS 3.0
6 Mb di RAM, HD 130 Mb



AMIGA 600
£. 479.000

Ram 1Mb, Clock a 32 bit, Interfaccia
HD incorporata
Modulatore PAL incorporato

Con HD 47 Mb £. 839.000, in omaggio
"Super Tools Utility"



LE APPLICAZIONI SCIENTIFICHE SULL'AMIGA

In questo numero parliamo dei vantaggi e degli svantaggi dell'uso dell'Amiga per questo tipo di applicazioni. Ci soffermeremo in particolare sulle librerie matematiche

di Alfredo Distefano

Abbiamo spiegato nel precedente articolo cosa intendiamo per applicazione di tipo scientifico/matematico, elencando le caratteristiche principali di questa categoria di programmi. Abbiamo fatto anche una rapida carrellata sui prodotti già esistenti per Amiga in questo campo, sia commerciali che di pubblico dominio. Si sono poi viste le motivazioni che possono spingere a scrivere una propria applicazione e gli eventuali vantaggi che ne derivano rispetto all'uso di un prodotto già fatto. Si è affermato che, disponendo degli opportuni strumenti, non bisogna essere né esperti programmatori né grandi luminari della scienza per ottenere dei risultati soddisfacenti.

Ora cercheremo di analizzare quali sono i vantaggi o gli svantaggi di sviluppare applicazioni scientifiche su una macchina come l'Amiga. Quando ci si accinge a sviluppare un'applicazione, infatti, si è inevitabilmente influenzati in positivo o in negativo dalle caratteristiche della macchina utilizzata. È utile quindi valutare su quali potenzialità si può contare e a quali eventuali manchevolezze bisogna porre rimedio per riuscire a impostare bene l'architettura del programma.

Eviteremo nel seguito dell'articolo di descrivere in dettaglio quali sono le caratteristiche hardware e software dell'Amiga: quei lettori che non le conoscessero sono invitati a leggere i numerosi articoli già pubblicati in proposito su questa rivista. Si cercherà invece di valutare come alcune di queste caratteristiche possano influire su un'applica-

zione di tipo scientifico/matematico. Analizzeremo in particolare il problema della rappresentazione dei numeri decimali.

L'hardware dell'Amiga

I coprocessori custom costituiscono sicuramente la caratteristica più inconfondibile dell'Amiga, quindi è inevitabile iniziare le nostre considerazioni a partire da questi. Con gli ultimi modelli della Commodore, Amiga 1200 e 4000, la famiglia di questi chip si è molto arricchita e potenziata, a tal punto che è diventato difficile fare un discorso generale su di essi. Rimane però inconfutabile che, al di là delle versioni di chip custom installate sulla macchina, la loro presenza permette di avere buone, anzi ottime, prestazioni grafiche e sonore anche con CPU ormai non particolarmente potenti come un 68000 a 8 MHz. Nell'ottica del nostro discorso, è quindi chiaro che le applicazioni scientifiche o matematiche che più trarranno vantaggio da questa caratteristica sono quelle che prevedono una parte grafica o che addirittura rientrano nella categoria dell'elaborazione digitale delle immagini. Sfruttando bene queste risorse, infatti, è possibile scaricare molta dell'elaborazione grafica sui chip custom, lasciando abbastanza libera la CPU per effettuare i calcoli veri e propri.

Il discorso è un po' diverso per quanto riguarda le caratteristiche sonore, che non sembrano portare molti vantaggi a un'applicazione scientifica. Potrebbero però venir sfruttate per applicazioni che

si occupino per esempio dello studio della sovrapposizione di onde sonore; se qualche lettore ha segnalazioni da farci su quest'argomento, è invitato a presentarle in redazione.

Anche se l'applicazione non è di tipo grafico o sonoro, comunque, la presenza dei chip custom aiuta molto nella gestione di un'interfaccia uomo-macchina che risulti efficace e comoda da usare, senza per questo dover sottrarre molti cicli di CPU della macchina per la gestione di finestre o menu.

Continuando a parlare di applicazioni scientifiche che coinvolgono aspetti di grafica, l'Amiga presenta il vantaggio che da quando è stato presentato sul mercato dispone di un controller grafico direttamente sulla scheda madre. Anche in questo caso, l'evoluzione dei chip ha portato molti miglioramenti rispetto al passato, ma in generale un qualsiasi programmatore può contare sul fatto che le risoluzioni standard dell'Amiga saranno sicuramente utilizzabili su qualsiasi modello Amiga, senza doversi preoccupare di controllare o richiedere la presenza di hardware aggiuntivo.

Questo aspetto ha anche il suo lato negativo: mentre è vero che le risoluzioni standard sono a disposizione di tutti, è anche vero che queste risoluzioni pongono dei limiti che in alcuni casi possono costituire un problema. Se non teniamo conto dei nuovi chip AGA, per applicazioni di studio dei frattali, per esempio, potrebbe risultare molto limitante il massimo di 16 colori per gli schermi ad alta risoluzione. Un altro

problema può nascere per coloro che non dispongono di monitor multisync, o perlomeno VGA, per i quali l'utilizzo delle risoluzioni con interlace può risultare molto fastidioso a causa dello sfarfallio. Se quindi un'applicazione richiede certi tipi di grafica, il programmatore si trova costretto a scegliere se limitare l'uso del programma ai pochi eletti che dispongono dell'ultima versione dei chip custom oppure ridurre i propri requisiti grafici a spese della qualità del risultato. La soluzione migliore, in questo caso, sarebbe il supporto da parte del programma di tutti i possibili modi grafici, in modo che l'utente possa scegliere quali utilizzare a seconda delle caratteristiche della macchina, ma spesso questa scelta implica uno sforzo di programmazione che non tutti si sentono di sostenere.

Per questo tipo di problemi, va cercata una soluzione caso per caso; cerchiamo comunque di dare alcuni consigli. Se per la vostra applicazione ritenete necessario utilizzare l'interlace, cercate di trovare una combinazione di colori che minimizzino l'effetto di sfarfallio su un monitor non VGA. Cercate inoltre di scegliere delle fonti che risultino di facile lettura, senza però fare l'errore opposto di richiedere che l'utente abbia installata nel suo sistema una fonte estremamente esotica solo perché vi sembra esteticamente accattivante: ricordatevi che lo scopo ultimo di un'applicazione scientifica non è l'estetica.

Se la vostra applicazione richiede l'utilizzo di molti colori, cercate di non limitarne l'uso ai soli possessori di chip AGA, che almeno per ora non sono molti. Cercate d'altra parte d'includerne il supporto, anche se la macchina su cui sviluppate non li monta: è molto deludente infatti vedere un programma che non sfrutta tutte le potenzialità di una macchina proprio dove ne avrebbe più bisogno.

Parliamo ora di quello che è il vero cuore della macchina, la CPU. È noto che tutti i modelli Amiga sono basati sulla famiglia dei processori Motorola 680X0. Questi processori, rispetto ai concorrenti, hanno degli innegabili vantaggi per quanto riguarda la facilità di gestione della memoria e in generale per quanto riguarda la gestione dei dati, grazie alla loro rappresentazione di tipo *Big Endian*, cioè con il byte più significativo a sinistra. Queste caratteristiche, anche se i fan dei processori Intel

potrebbero non essere d'accordo, facilitano la programmazione anche di applicazioni scientifiche, soprattutto se c'è bisogno di gestire una grande quantità di dati in memoria.

Quello che però soprattutto importa per un'applicazione scientifico/matematica, è la prestazione in termini di capacità di calcolo del processore, con particolare riguardo ai calcoli in virgola mobile e all'utilizzo di funzioni trascendenti (seno, coseno, eccetera). Da questo punto di vista, le caratteristiche delle macchine Amiga variano moltissimo da modello a modello. Gli Amiga di base dotati di un semplice 68000 a 8 MHz sono seriamente svantaggiati, anche se la CPU è coadiuvata dai coprocessori grafici. Si ha già un sensibile salto di qualità in macchine dotate di 68020, come l'Amiga 1200, che oltre ad avere

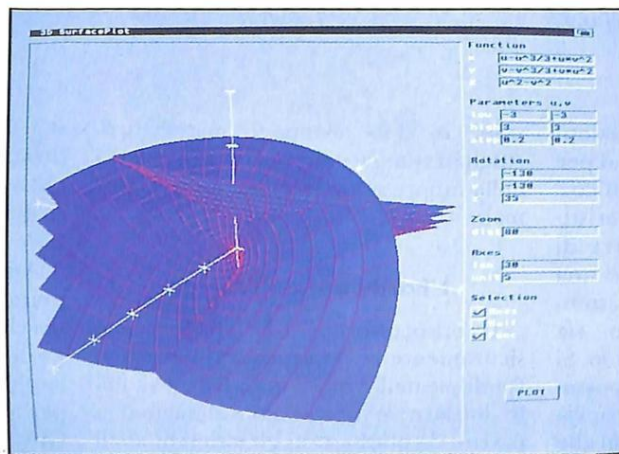
sore; chi possiede macchine con 68000 può ricorrere per esempio alle schede prodotte dalla Hardital, che con una spesa intorno alle 350 mila lire permettono di dotare la propria macchina di 68020 a 16 MHz con coprocessore 68882 a 25 MHz. Per i possessori di Amiga 1200, sempre la Hardital esegue l'installazione del coprocessore sulla scheda madre per circa 150 mila lire; un'altra soluzione (che non invalida la garanzia) è quella di acquistare una scheda come la Microbotics MBX 1200 che con il 68881 a 14 MHz costa circa 370 mila lire.

Una piccola nota sulla differenza tra i coprocessori 68881 e 68882: l'insieme d'istruzioni supportate dai due coprocessori è identico, solo che il 68882 ha un'architettura interna parzialmente parallela che permette un minor tempo medio di esecuzione della singola istruzione. Purtroppo, però, per poter sfruttare al meglio questa potenzialità bisognerebbe che il compilatore usato tenesse conto di queste differenze tra i due coprocessori, cosa che normalmente non avviene.

La CPU 68040, come quella montata nell'Amiga 4000, merita un discorso a parte, perché in questo caso il coprocessore matematico è integrato nella CPU stessa. Questa soluzione dovrebbe teoricamente far guadagnare enormemente in termini di prestazioni, ma non è del tutto vero: la Motorola, infatti, per integrare il coprocessore in un unico chip ha

dovuto sacrificare molte delle istruzioni originali. Di fatto, l'unità integrata nel 68040 permette di fare direttamente solo le operazioni fondamentali, mentre tutte le altre, come le funzioni trascendenti, vengono emulate via software: nonostante questo, le prestazioni dal punto di vista matematico del 68040 sono comunque leggermente superiori a quelle dell'accoppiata 68030 + 68882, a parità di clock.

Tutte queste considerazioni, che possono tornare utili se si vuole potenziare la propria macchina per scopi di tipo scientifico, non devono scoraggiare chi non possiede il coprocessore e non ha la possibilità di ottenerlo. Come vedremo anche dal nostro esempio di applicazione, infatti, si può creare il programma in modo tale che se nel sistema esiste un coprocessore matematico, venga sfruttato per i calcoli con un grande incremento delle prestazioni, ma se il coprocessore è assente le sue funzionalità



Surface Plot, programma matematico di pubblico dominio che fa uso di librerie già pronte per gestire interface utente

un clock generalmente di 16 MHz possiedono un bus dati a 32 bit e una cache istruzioni, rispetto ai 16 bit e nessuna cache del 68000. Ancora meglio da questo punto di vista sono i 68030, montati sull'Amiga 3000, che hanno una cache anche per i dati e generalmente hanno un clock di 25 MHz. Questi ultimi due processori, poi, possono essere affiancati dal coprocessore matematico, il 68881 o meglio ancora il 68882, che per le applicazioni matematiche rappresenta un enorme miglioramento, sicuramente più importante di quello che si ottiene cambiando solo la CPU. Tale coprocessore, infatti, è specializzato proprio nell'esecuzione dei calcoli in virgola mobile e nell'esecuzione di funzioni trascendenti, che così vengono eseguite molto velocemente e parallelamente al lavoro svolto dalla CPU principale. Se quindi si è interessati alle applicazioni scientifiche, è molto consigliabile dotarsi di questo coprocessore.

vengano perfettamente emulate dal software. Anche se in questo secondo caso l'esecuzione risulterà sensibilmente più lenta, questa soluzione permette di sviluppare e di eseguire l'applicazione su qualsiasi macchina.

Il software dell'Amiga

Il software di sistema dell'Amiga può essere considerato di ottimo livello, soprattutto dalla versione 2.0 in poi. Anche se può sembrare strano, la capacità del sistema di supportare il multi-tasking risulta utile nelle applicazioni di tipo scientifico/matematico. Molto spesso, infatti, il programma deve continuamente eseguire calcoli anche piuttosto pesanti per far evolvere la simulazione, ma nello stesso tempo dev'essere in grado di gestire l'interfaccia verso l'utente per permettergli per esempio d'intervenire sui parametri di simulazione. Ora, mischiare nello stesso codice la gestione dell'interfaccia utente insieme con il ciclo di ricalcolo può risultare difficoltoso; inoltre, la gestione dell'interfaccia dovrebbe entrare in gioco solo quando è necessario, cioè quando l'utente fa qualche azione per intervenire, mentre per la maggior parte del tempo dev'essere il calcolo a occupare la CPU, possibilmente senza dover continuamente controllare le azioni dell'utente. Sfruttando la possibilità del multi-tasking, è possibile scrivere un codice molto leggibile, in cui la parte di gestione dell'interfaccia e quella di calcolo vero e proprio sono praticamente indipendenti e organizzati in modo tale da sfruttare al meglio la potenza della CPU a seconda delle necessità.

Vi sono anche altri motivi per cui è conveniente riuscire a dividere la parte di gestione dell'interfaccia da quella di calcolo. Quest'ultima, infatti, essendo il vero cuore del programma, potrà essere particolarmente ottimizzata per migliorarne le prestazioni. L'interfaccia verso l'utente, invece, non deve necessariamente essere molto ottimizzata, in quanto di norma viene utilizzata solo saltuariamente e per breve tempo rispetto a quello occupato dal calcolo. Inoltre, al variare nel tempo dei gusti o degli strumenti di sviluppo a nostra disposizione, può risultare utile modificare, arricchire o addirittura sostituire completamente la parte d'interfaccia, senza per questo dover toccare una sola riga della parte di calcolo.

A tutto questo si aggiunge il fatto che spesso la parte di calcolo viene estratta da un testo di riferimento, come dicevamo nel precedente articolo. Se dunque questa parte di codice rimane separata

dal resto del programma, si può riuscire anche a mantenerla praticamente uguale all'esempio da cui è tratta, facilitando in questo modo la verifica della correttezza dell'implementazione.

Il software di sistema dell'Amiga, d'altra parte, facilita enormemente la gestione dell'interfaccia verso l'utente. La libreria di "Intuition", insieme alle nuove librerie "asl" e "gadtools" presenti dalla versione 2.0 in poi, permettono con relativa facilità di sviluppare interfacce molto comode da usare per l'utente e molto facili da gestire per il programmatore. Il fatto poi che queste librerie facciano largo uso dei coprocessori custom fa sì che molta della gestione di questa interfaccia non sottragga la CPU alla parte di calcolo che, ricordiamo, deve avere la precedenza in un'applicazione scientifico/matematica. Sarebbe infatti molto seccante se, per realizzare un'interfaccia gradevole, quasi metà del tempo di esecuzione della CPU fosse impiegato per gestirla, magari solo perché l'utente sta spostando una finestra o ha abilitato un menu. È confortante invece sapere che la maggior parte di queste operazioni verranno scaricate sui coprocessori e che il nostro programma verrà avvertito dal sistema solo quando l'utente ha effettuato una vera e propria scelta. A questo punto il nostro programma non dovrà fare altro che eseguire la funzione corrispondente a questa scelta.

Un altro elemento molto positivo per lo sviluppo di un'applicazione sull'Amiga è dato dalla completezza delle librerie di sistema, per cui spesso per eseguire determinate operazioni non è necessario scrivere del proprio codice, ma è sufficiente passare i giusti parametri alle funzioni di sistema opportune. Questo vale anche per l'aspetto squisitamente matematico, in quanto le librerie di sistema comprendono anche la gestione di tutti i calcoli con variabili in virgola mobile. In questo modo, indipendentemente dal linguaggio o dal compilatore usato, è sempre possibile effettuare operazioni matematiche di un certo livello con relativa facilità. Queste librerie, inoltre, controllano automaticamente la presenza del coprocessore matematico: se c'è, lo usano, se non c'è, ne emulano le funzioni via software. Il codice resta così compatibile con tutte le macchine, ma contemporaneamente ne sfrutta a fondo le capacità.

Il linguaggio C

Il linguaggio che abbiamo scelto per lo sviluppo del programma di esempio di questa serie di articoli è il linguaggio

C. Questa scelta è praticamente obbligata per vari motivi: innanzitutto perché tale linguaggio è in assoluto il più utilizzato sull'Amiga. Questo porta come immediata conseguenza che nel pubblico dominio è facile trovare ottimi programmi anche in campo scientifico, come avevamo già fatto notare nel precedente articolo, corredati spesso dai sorgenti in C che possono essere utilizzati come modelli per le proprie applicazioni.

Sempre nel pubblico dominio è facile anche trovare molte librerie di aiuto alla programmazione in linguaggio C, utilissime per facilitare la gestione dell'interfaccia verso l'utente o per effettuare operazioni particolari senza volersi scrivere da soli tutte le routine. Una di queste librerie, la "reqtools" di Nico Francois, è stata utilizzata anche per il programma di esempio. Tale libreria semplifica enormemente la gestione dei requester, permettendo di sviluppare comode interfacce con poche chiamate di funzione.

Accanto alle librerie per il linguaggio C, nel pubblico dominio si possono trovare anche dei veri e propri generatori di codice che aiutano il programmatore nella scrittura di certe parti del programma. Un esempio di questo tipo di programmi è "gadtoolsbox", anch'esso impiegato nello sviluppo del programma di esempio. Mediante questa utility è possibile "disegnare" sullo schermo l'aspetto dell'interfaccia, determinando per esempio il tipo di schermo usato, i colori, l'aspetto delle finestre, i menu... Una volta finito il disegno, il programma genera automaticamente il codice C comprendente le funzioni di creazione, cancellazione e gestione dell'interfaccia. A noi non resta che aprire le librerie opportune e scrivere le funzioni corrispondenti alle selezioni che l'utente può fare.

Tutto ciò è semplificato dal fatto che lo stesso software di sistema dell'Amiga è scritto in C e che quindi l'utilizzo di questo linguaggio risulta "naturale"; la Commodore, infatti, fornisce gli header in linguaggio C contenenti tutte le strutture e le definizioni di funzione necessarie per utilizzare immediatamente le librerie di sistema.

Come si può capire da queste considerazioni, l'impiego del linguaggio C, anche se può apparire più disagiata per chi non conosce bene questo linguaggio, risulta alla fine molto facile da utilizzare grazie alle numerose utility e librerie già pronte.

Un altro vantaggio per chi programma in linguaggio C è che esistono per Amiga degli ottimi compilatori. Nel

campo del pubblico dominio c'è il compilatore *DICE*, molto utilizzato per lo sviluppo di utility, mentre per quanto riguarda i prodotti commerciali c'è il compilatore *SAS/C*, giunto ormai alla versione 6.2. Quest'ultimo in particolare presenta molte caratteristiche avanzate, come la generazione automatica dei prototipi delle funzioni o l'utilizzo del sistema *AmigaGuide* per gli help di tutti i comandi, che rendono più agevole lo sviluppo di programmi anche di grandi dimensioni. La presenza inoltre di un buon ottimizzatore del codice permette di ottenere programmi finali molto veloci: questa caratteristica può risultare importante per programmi che devono eseguire pesanti calcoli nel più breve tempo possibile.

Le librerie matematiche

Parlando di programmi di tipo scientifico, è importante valutare come il linguaggio di programmazione scelto gestisca i calcoli matematici; bisogna per esempio conoscere con quale precisione possono essere rappresentati i numeri decimali. Da questo punto di vista il linguaggio C non è mai stato considerato il miglior linguaggio utilizzabile: spesso le applicazioni matematiche venivano scritte infatti in linguaggio Fortran che, almeno in passato, assicurava una miglior precisione nei calcoli.

Col passare del tempo, anche il linguaggio C ha subito un'evoluzione da questo punto di vista; sull'Amiga poi, grazie anche alle librerie matematiche fornite dalla Commodore con il sistema, la situazione è ancora migliore. Per evitare però di commettere errori che rischierebbero d'invalidare completamente i risultati ottenuti dalla nostra applicazione, è opportuno esaminare in dettaglio l'argomento.

Il linguaggio C permette due tipi fondamentali di variabili in virgola mobile: il primo tipo, denominato *float*, rappresenta un numero su 32 bit, mentre il secondo, denominato *double*, rappresenta un numero su 64 bit. Il numero di bit utilizzati, però, non è sufficiente per determinare la precisione raggiunta: bisogna anche tener conto del formato di rappresentazione delle variabili in virgola mobile, che non è univoco come si potrebbe pensare. Sull'Amiga si pos-

sono utilizzare due formati diversi: il "Motorola Fast Floating Point", o FFP, e il formato "IEEE".

Il formato FFP rappresenta i numeri su 32 bit e quindi va usato solo con variabili dichiarate *float*. La caratteristica di questo formato è che permette di effettuare calcoli abbastanza velocemente, pur non possedendo un coprocessore matematico, a scapito però della precisione raggiunta. Per l'esattezza, gli intervalli di valori rappresentabili vanno per i numeri positivi da 9.22337177E18 a 5.42101070E-20 e per i numeri

sufficiente specificare l'opzione *math=ffp*.

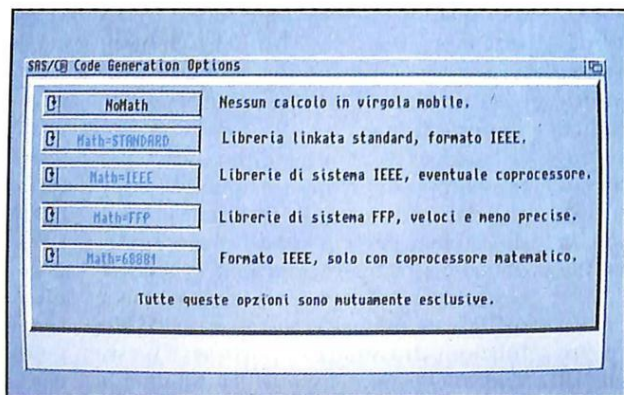
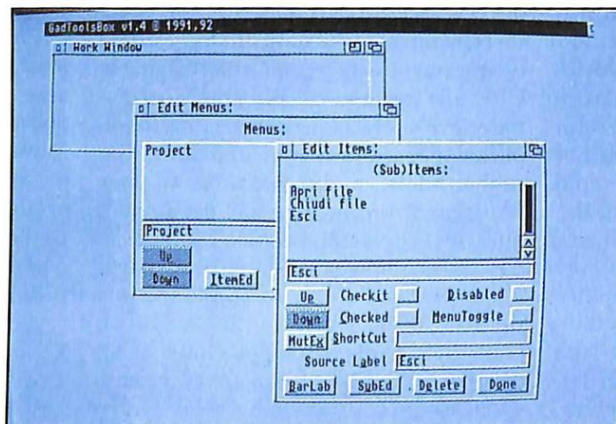
Il formato "IEEE" è a sua volta suddiviso in due sotto-casi: il formato in singola precisione, su 32 bit, e quella in doppia precisione, a 64 bit. Nel primo caso l'intervallo di valori permesso è da 1.3E-38 a 3.4E38; come si vede, a parità di bit usati (32) questo formato è molto più preciso di quello FFP. Questa precisione viene purtroppo pesantemente pagata in termini di velocità di calcolo, ma questo però è anche il formato utilizzato internamente dai coprocessori matematici della Motorola. L'utilizzo quindi di questo formato con un coprocessore matematico unisce una grande precisione di calcolo con una grande velocità di esecuzione.

Il formato "IEEE" a doppia precisione riesce a rappresentare numeri da 2.2E-308 a 1.8E307 (cioè 1 seguito da ben 307 zeri!). Per questo tipo di formato valgono le stesse considerazioni fatte per quello in singola precisione; naturalmente, dato il più alto numero di bit coinvolti nei calcoli, a parità di formule utilizzate il formato doppio sarà più lento di quello singolo.

Tutti e due i tipi di formati "IEEE" sono utilizzabili con le librerie di sistema fornite dalla Commodore. Tali librerie, come abbiamo già detto in precedenza, si occupano tra l'altro di controllare la presenza del coprocessore matematico e, se c'è, lo utilizzano. Per utilizzare queste librerie con il compilatore *SAS/C* 6.2, è sufficiente specificare l'opzione *math=ieee*. Il compilatore *SAS/C* permette di utilizzare il formato "IEEE" anche con le proprie librerie, che non fanno uso di quelle di

sistema. In particolare, se si specifica il parametro di compilazione *math=standard*, verrà utilizzata la libreria linkata "scm.lib". Tale libreria dovrebbe permettere calcoli leggermente più veloci di quelli permessi dalle equivalenti librerie di sistema in assenza del coprocessore matematico; d'altra parte, non sono in grado di utilizzare il coprocessore se presente. Si tenga inoltre conto che essendo una libreria linkata, e non di tipo "shared" come quelle di sistema, il programma risulterà di dimensioni molto più grandi.

Un altro tipo di parametro di compilazione che può essere specificato con il *SAS/C* è *math=68881*. In questo caso il



Sopra: esempio di definizione di un menu con l'utility GadToolsBox. Sotto: parametri di compilazione del SAS/C sulle librerie matematiche (immagine elaborata con PPaint)

negativi da -9.22337177E18 a -2.71050535E-20. Per chi non ha dimestichezza con la notazione scientifica, E18 significa che al numero precedente vanno aggiunti 18 zeri, mentre E-20 significa che il numero dev'essere preceduto da 20 zeri e tra il primo e il secondo va posto il punto decimale. Per esempio, 1E3 indica il numero 1000 (1 seguito da tre "0"); 1E-3 indica il numero 0.001 (1 preceduto da tre "0"). Il formato FFP è implementato mediante due librerie di sistema, una per le operazioni matematiche di base (addizione, sottrazione...) e una per le operazioni trascendenti (seno, coseno...). Per utilizzare questo formato con il compilatore *SAS/C* 6.0, è

compilatore per i calcoli in virgola mobile chiamerà direttamente le istruzioni del coprocessore matematico, senza passare attraverso le chiamate di sistema. Questo metodo è in assoluto il più veloce se si è sicuri di poter usufruire del coprocessore matematico, soprattutto se nei sorgenti che fanno uso di calcoli in virgola mobile s'incluse il file "m68881.h", ma il programma così ottenuto non funzionerà su sistemi privi del coprocessore.

Questa analisi sui formati di rappresentazione dei numeri decimali e sulle librerie relative potrà sembrare un po' complessa, ma è un argomento da non sottovalutare in fase di sviluppo di un'applicazione scientifico/matematica e lo mostreremo con un esempio. Supponete di scrivere un programma di argomento astronomico, come quello che presenteremo nel prossimo articolo, e che nei calcoli si debba fare uso della massa del Sole. Se come unità di misura delle masse si usano i grammi, cioè quella normalmente utilizzata dagli astrofisici, si troverà che la massa del Sole è pari a 2E33 grammi (cioè 2 seguito da 33 zeri). Se confrontate questo valore con gli intervalli di valori ammessi per ciascun formato che abbiamo prima elencato, potete capire che se compilate il vostro programma utilizzando il formato FFP il risultato di qualsiasi calcolo che contenga la massa del Sole sarà completamente privo di significato e così l'intero programma servirà a ben poco. Se invece si fa uso del formato "IEEE" a singola precisione, la massa del Sole potrà essere correttamente rappresentata, ma si dovrà porre in ogni caso attenzione alle formule

utilizzate. Se per esempio in qualche formula tale massa viene elevata al quadrato, o viene moltiplicata per un valore altrettanto grande, di nuovo si andrà oltre l'intervallo di valori rappresentabili e il calcolo perderà di significato. Si avrebbero invece molti meno problemi utilizzando il formato "IEEE" a doppia precisione.

Un problema del tutto simile a quello dei grandi numeri si ha con i numeri molto piccoli, oppure con quelle formule in cui piccolissime variazioni di una variabile portano a grandi cambiamenti dei risultati: in tutti questi casi è sempre preferibile avere la massima precisione possibile e usare quindi il formato "IEEE" a doppia precisione, almeno inizialmente. Una volta verificati i risultati ottenuti in questo modo, si potrà provare a utilizzare diversi formati, come il FFP, confrontando però i risultati così ottenuti con quelli precedenti. Può darsi che le differenze siano trascurabili e che quindi possiate utilizzare indifferente i due formati. Attenzione però che le differenze tra un formato e l'altro, se trascurabili per certi valori dei parametri di calcolo, potrebbero diventare invece critiche per altri valori dei parametri: state quindi sempre bene attenti agli intervalli numerici usati.

In qualche caso non è tanto necessario aumentare la precisione sui calcoli, quanto riuscire a scegliere la giusta unità di misura, soprattutto nei calcoli di tipo fisico. Tornando all'esempio del programma astronomico, se si devono effettuare dei calcoli che coinvolgono le distanze dei pianeti, e si usa come unità di misura il centimetro, si avrebbe subito a che fare con numeri enormi. Se

invece si decide di usare come unità di misura la distanza media tra Terra e Sole, che in astronomia viene chiamata U.A. (Unità Astronomica), anche la distanza di un pianeta lontano come Saturno, per esempio, può essere indicata con numeri relativamente piccoli. È quindi importante decidere anche il giusto sistema di unità di misura da usare, almeno all'interno delle formule: una volta ottenuto il risultato sarà sempre possibile trasformarlo in qualsiasi unità di misura si desideri.

Si ricordi anche quello che si era già accennato nell'articolo precedente: le unità di misura utilizzate dovranno sempre essere coerenti fra di loro. Se si utilizzano per esempio i metri per indicare una distanza, e le U.A. per indicarne un'altra, e se ne moltiplicano i valori numerici tra di loro, il risultato non sarà esprimibile né in metri né in U.A.: sarà prima necessario trasformare una distanza nell'unità di misura dell'altra. Potrete trovare su un qualsiasi buon libro di testo tutte le formulette che vi interessano per passare da un'unità all'altra.

In questo articolo e nel precedente abbiamo analizzato i problemi principali in cui s'imbatte un programmatore di applicazioni scientifiche sull'Amiga e abbiamo suggerito qualche possibile soluzione. Nel prossimo articolo presenteremo un esempio di applicazione di carattere astrofisico, in cui tutti i concetti fin qui esposti saranno messi in pratica. Rinnoviamo l'invito a tutti i lettori che hanno scritto applicazioni scientifico/matematiche di spedire in redazione i loro lavori o di porci eventuali quesiti sull'argomento. ■

La Compugraph è lieta di presentare:

Il Workbench facile

...ovvero il primo corso in videocassetta che vi insegnerà in modo semplice ed efficace ad utilizzare al meglio l'ambiente Workbench sui computers Commodore della serie Amiga. Primo nel suo genere il corso prende per mano sia il neofita che l'utente esperto e li accompagna nel lungo viaggio attraverso menu, dischi, cassette e programmi inclusi nel dischetto Workbench 2.0. Il corso della durata di 2 ore e 30 minuti è corredato di numerosissimi esempi pratici, schermate esemplificatrici ed animazioni. Tutto questo rende il prodotto un mezzo estremamente versatile ed efficace in grado di portare l'utente Amiga ad una buona conoscenza del Workbench versione 2.04. La videocassetta <Il Workbench Facile> è disponibile presso i rivenditori specializzati di tutta Italia al prezzo di Lit. 39.900.

Videocassetta Hi-Grade VHS PAL durata 2 ore e 30 min

DISTRIBUZIONE A CURA DELLA RAINBOW COMPUTING S.A.S. GENOVA - TEL. (010) 58.44.25 FAX (010) 58.44.26

SI CERCANO RIVENDITORI PER ZONE SCOPERTE

LO SPAZIO DI LYAPUNOV OVVERO LA BELLEZZA DEL CAOS

Esploriamo con il nostro Amiga uno degli insiemi matematici più affascinanti e complessi

di Antonio De Lorenzo

Che per un matematico la propria materia possieda una coerenza e una bellezza interna è a volte incomprensibile al grande pubblico, ma assolutamente innegabile. In alcune branche della matematica tale bellezza si rivela esterna a quello che può essere il rigore formale, l'eleganza di una dimostrazione, il rimando e la simmetria interna, la sintesi, eccetera. L'introduzione dei calcolatori per usi matematici non ha permesso solo l'accelerazione di calcoli, la verifica e dimostrazione di equazioni e sistemi complessi, ma ha consentito nel contempo la rappresentazione grafica di funzioni e insiemi spalancando di fatto universi interi e portando la matematica stessa a frontiere inimmaginabili fino a una decina di anni fa.

La grafica al calcolatore rivela la bellezza insita in molti processi matema-

tici e fisici consentendone una fruizione estetica anche ai "non iniziati". Gli esempi più noti riguardano l'attrattore di Lorentz, gli insiemi di Mandelbrot e Julia, i frattali... che hanno consentito al

nel quale la strana geometria "aliena" e le leggi che lo governano ne fanno un altro insieme bellissimo da esplorare e ammirare. Potreste chiedervi quale sia il senso di tutto ciò, del resto queste

restano delle semplici immagini. In conclusione d'articolo, vedremo come tale tipo di spazio consenta applicazioni molto avanzate al di fuori del significato specialistico. Tenete poi conto che questo tipo d'immagini cerca di dare finalmente un volto alla "quantità" di disordine, al caos da sempre concetto sfuggitivo e complesso da comprendere e definire. Le immagini ottenute possono essere impiegate in modi diversi ma anche solo così, semplicemente da ammirare, giu-



Un'immagine dello spazio di Lyapunov generata impiegando Lyapunovia versione 1.0

grande pubblico di apprezzare universi irreali, lontani e fantasmagorici, regolati da leggi proprie e geometrie interne d'indubbia bellezza e fascino. È successo nuovamente con lo spazio di Lyapunov,

stificano il tempo speso a ricercarle e scoprirle. Fanno parte di ciò che alcuni chiamano "l'indispensabile cibo per la mente". E dobbiamo proprio dire che è un cibo davvero invitante...

Cos'è lo spazio di Lyapunov

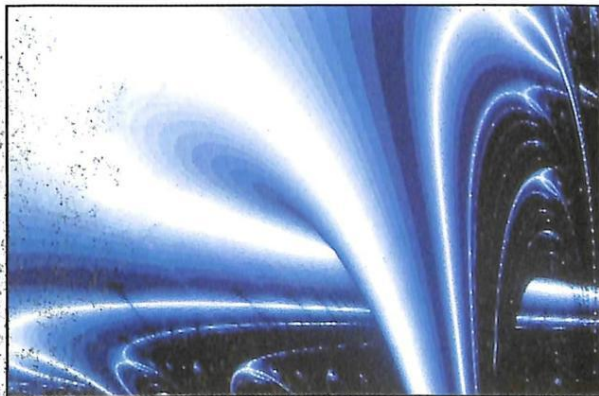
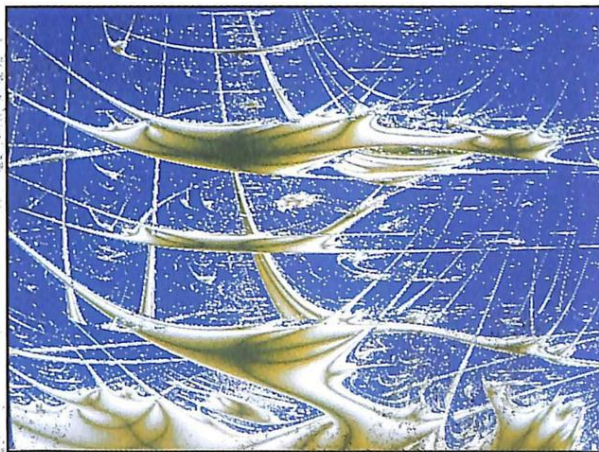
Tutto ebbe inizio nei laboratori del famoso Max Planck Institut für Ernährungsphysiologie (Istituto per la nutrizione) a Dartmond, Germania. Un ricercatore, Mario Markus, insieme al suo collaboratore Benno Hess studiavano dei modelli matematici per simulare il modo con cui certi enzimi (proteine specializzate considerate dei catalizzatori altamente specifici) demoliscono alcuni componenti fondamentali della nostra dieta come i carboidrati. Intervendendo su una coppia di parametri, i due ricercatori avevano notato che potevano determinare un comportamento caotico od ordinato di tali enzimi. Per quanto la cosa vi possa meravigliare, queste immagini danno proprio la misura del loro comportamento caotico. Tutte le immagini generate si basano su una formula detta per l'appunto di Lyapunov, dal nome del matematico russo Aleksandr M. Lyapunov, che genera un solo numero per un qualsiasi sistema dinamico. Questo numero è chiamato esponente di Lyapunov ed è indice del comportamento caotico dell'intero sistema. Ora applicando la formula di Lyapunov al modello sviluppato dai due ricercatori, questa produce un esponente per ogni coppia di parametri. Per visualizzare tale processo si assegna per ogni coppia di valori un punto sullo schermo e si attribuisce loro un colore a seconda della grandezza di tale esponente. In questo modo si generano immagini dello spazio di Lyapunov. Tenete presente che in queste immagini l'ordine disegna le forme in chiaro mentre il disordine quelle scure.

Se l'insieme di Mandelbrot è stato definito "l'oggetto più complesso in matematica", allora lo spazio di Lyapunov è senz'altro l'oggetto più esteso, interessante ed eterogeneo mai trovato per mezzo di numeri. Le immagini provenienti da questo insieme sono così belle che l'autore, dopo aver pubblicato le sue ricerche su riviste accademiche, ha pensato bene d'inviarle ad alcune gallerie d'arte perché le esponessero. Come potrete convenire, queste immagini hanno sembianze di quadri astratti che potrebbero benissimo essere state stese su tela da un qualche allievo di Salvador Dalí, piuttosto che essere state

generate da una formula matematica sullo schermo di un calcolatore grafico.

Esploriamo lo spazio di Lyapunov con Lyapunovia

Il mondo dei programmatori Amiga è attentissimo alle novità e alle nuove conquiste grafiche (prova ne è che disponiamo dei migliori programmi 3D in assoluto) e al solito per primi per i nostri calcolatori si sono resi disponibili



Sopra: una suggestiva immagine creata da Lyapunovia, un ottimo programma shareware di esplorazione dello spazio di Lyapunov realizzato in Danimarca da Jesper Juul. Sotto: una schermata generata impiegando Lyapunov Fractals

programmi in grado di esplorare un territorio tanto affascinante quanto immateriale. Tra i programmi, rigorosamente PD e shareware, per l'esplorazione dell'insieme, ne abbiamo scelti due facilmente reperibili e molto sofisticati. Esaminiamoli.

Lyapunovia 1.0 è un ottimo programma shareware di esplorazione dello spazio di Lyapunov proveniente della Danimarca, ne è autore un certo Jesper Juul. Su un unico disco è presente una gran quantità di materiale. Un manuale di quasi 30K in inglese, molto ben realizzato, già preimpaginato (basta stamparlo), con indice e un'ottima divisione degli

argomenti. Troviamo inoltre il programma principale che occupa solo 22K, undici file per personalizzare nella propria lingua il programma tra cui anche l'italiano, alcune palette di colori predefinite particolarmente indicate per i buoni accostamenti, coordinate e parametri per le zone più interessanti trovate dall'autore, una serie d'immagini tra le più significative.

Se avete già dimistichezza con i programmi che consentono di esplorare l'insieme di Mandelbrot con le varie e classiche opzioni di zoom, cambiamento e cycle della palette di colori... potrete facilmente comprendere come lavora *Lyapunovia*. Immaginatevi un quadrato riportante coordinate che vanno da 2.0 a 4.0 orizzontalmente e da 2.0 a 4.0 anche verticalmente. Questo è ciò che si vede quando si esegue il programma, mentre l'utente può sempre ingrandire per osservarne particolari e dettagli come nei programmi per l'esplorazione dell'insieme di Mandelbrot. Ciò che rende però *Lyapunovia* molto più affascinante da utilizzare, è la possibilità di scegliere tra un numero infinito di diversi domini, cioè zone con caratteristiche uniche che l'autore non esita a definire come zone o territori "aventi una personalità e un carattere proprio". E le immagini ogni volta si fanno strane, distaccate, come se fosse frutto di un'immaginazione non umana.

Comunque, lo spazio di Lyapunov non è una mappa del caos negli stessi termini dell'insieme di Mandelbrot. È comunque una meta-map. Come abbiamo visto, il programma calcola l'esponente di Lyapunov per ciascun pixel e indica se la formula è ordinata o caotica a una precisa posizione X, Y. Abbiamo visto che il caos è riportato in nero o in colori scuri, mentre l'ordine con colori più chiari. La formula di base è equivalente a una utilizzata per gli "Alberi di Feigenbaum": $x = rx(1-x)$. Si veda comunque la bibliografia finale per maggiori dettagli tecnici.

Lyapunovia parte generando una semplice immagine dello spazio. Il programma è predisposto per iniziare automaticamente i calcoli ogni volta che un qualunque parametro viene cambiato. È possibile partire nella ricerca da un'im-

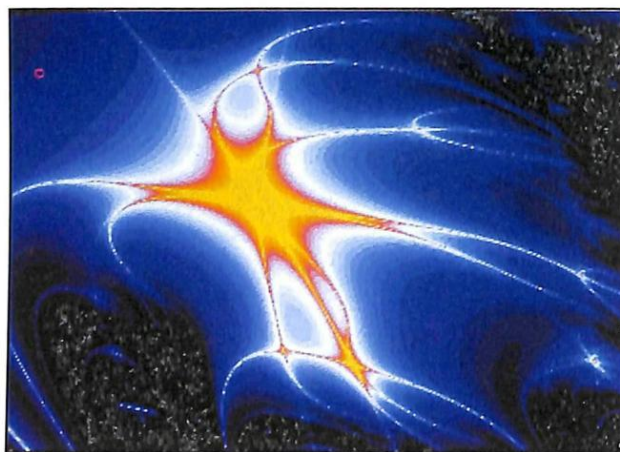
mage salvata (a patto che non sia stata modificata e riscritta con qualche programma di conversione formati o pittorico, altrimenti questa perde la parte aggiunta con le coordinate sotto forma di chunk TEXT), da una tra quelle fornite, da coordinate libere o anch'esse fornite per poi definirne accuratamente la palette. L'immagine ritenuta interessante può essere salvata in formato standard IFF. La ricerca avviene per zoom continui su zone prescelte. È possibile anche zoomare all'indietro (zoom in e zoom out). A mano a mano che comunque si scende per zoom continui, i tempi di calcolo aumentano e per zoom troppo estesi l'aspetto diviene a blocchi e non più ben definito. Questo è dovuto al fatto che la versione liberamente disponibile esegue calcoli in singola precisione. Un'aggiunta della versione registrata è proprio il supporto di calcoli in doppia precisione che consente di eseguire un numero d'ingrandimenti maggiori senza blocchi di pixel.

Il programma funziona su tutti i modelli di Amiga ed è stato testato con l'AmigaDOS in versione 1.3, 2.0 e 2.1, e richiede la ReqTools library in versione 2.0 (fornita) o superiore di Nico Francois (il programmatore di *Powerpacker*). È disponibile anche una collezione d'immagini che possono essere caricate all'interno del programma per continuare la ricerca di zone interessanti. Il programma come abbiamo detto è shareware, questo significa che se l'utente lo riterrà di suo gradimento e deciderà di utilizzarlo, è moralmente tenuto a spedire una piccola somma di denaro all'autore. Il programmatore chiede solo 50 corone, o \$15, e in cambio invierà una versione registrata ottimizzata per processori matematici e col supporto dei nuovi modi grafici a 256 colori dell'AGA. Inoltre, l'autore preannuncia che sta lavorando per adattare il programma a uscire direttamente su schede grafiche a 24 bit, per incrementarne ancora la velocità, inserire possibilità d'animazione, salvataggio in 24 bit dei file grafici e forse provvederà a implementare anche una porta ARexx.

Potete contattare direttamente l'autore scrivendo al seguente indirizzo: Jesper Juul, Guldmedgade 34, 1., DK-8000 Aarhus, Danimarca.

Lyapunov Fractals

Rimaniamo sempre in Europa con un altro programma PD di esplorazione dello spazio di Lyapunov, ma ci spostiamo questa volta in Francia. Il programma è stato scritto in F-Basic e poi compilato. I sorgenti acclusi si prestano a essere migliorati più facilmente vista la maggiore diffusione di questo linguaggio e la maggior leggibilità rispetto al C. Alcune routine in linguaggio macchina



Sopra e sotto: altre due suggestive immagini realizzate con *Lyapunov Fractals*, programma di PD scritto in F-Basic

rendono più veloce la gestione della palette e dell'overcan, oltre che di diverse altre funzioni. Da un esame veloce del listato, sembra anche che il programma sfrutti un eventuale processore matematico installato. Il tutto viene fornito in due floppy e funziona solo con O.S. 2.0 o superiore. Il programma è di qualità inferiore rispetto al precedente e anche più lento nell'esecuzione dei calcoli, ma come potete osservare dalle illustrazioni genera immagini molto belle. Manca un seppur minimo file di documentazione (pertanto non ne conosciamo neanche l'autore!), ma del resto il

tutto è molto semplice da utilizzare per mezzo dei soliti menu a tendina e del nostro fido mouse.

Cibo per la mente

A. K. Dewdney ha definito queste immagini: "un nutrimento per la mente e una festa per gli occhi". Effettivamente, aprono una porta su un universo tutto da esplorare e si fanno incontri ben più eccitanti di quanto non avvenga esplorando l'insieme di Mandelbrot e Julia. Si nota un'auto-somiglianza molto meno marcata e strane creature prendono a popolare il monitor se vi ci si dedica un po' del proprio tempo. I primi che esplorando nuovi spazi scoprono strutture non ancora esplorate possono conferire a questo una denominazione battezzando di fatto la propria scoperta così come avviene in campo geografico. Si conoscono già strutture ribattezzate "Rondine" o "Medusa" o ancora "Zircon city". Queste immagini possono essere utilizzate come brush per le operazioni di mapping all'interno di programmi 3D o come fondali e motivi per le proprie immagini 2D. Contribuiscono a dare un'atmosfera distaccata, aliena, tecnologica.

Personalmente, sono particolarmente grato all'autore di *Lyapunovia* dal momento che mi sono servito del programma per ricercare le immagini mappate sul pavone e sullo sfondo di *Lost in the Beauty of Lyapunov's Space*, l'immagine che ha vinto la selezione per il manifesto del Bit Movie di quest'anno e che avete potuto vedere sulla copertina del numero 1/93.

Bibliografia

- A. K. Dewdney, "Lo spazio di Lyapunov" - Rubrica L'angolo matematico su *Le Scienze* numero 279, novembre 1991, pp. 118-120. Markus Mario, "Chaos in Maps with Continuous and Discontinuous Maxima" su *Computers in Physics*, pp.481-493, settembre/ottobre 1990.

ULTIME NOTIZIE DAGLI USA

QUI NEW YORK, STATI UNITI

Tutto sui CD-ROM per Amiga, Art Expression e TypeSmith 1.0 della Soft-Logik, alla scoperta del DCTV, Deluxe Paint IV AGA...

di Morton A. Kvelson

Il compact disc è lo strumento ideale per la distribuzione di grandi quantità di dati digitali. La sua capacità supera i 600 MB il che nella maggior parte dei casi è più che adeguato per una grande varietà di applicazioni. Dal momento che i CD vengono stampati meccanicamente, il costo di produzione di un CD a sola lettura è basso: circa 2 dollari a disco per le prime 2 mila copie. Quando un'applicazione è così estesa da occupare una dozzina di floppy disk, è quindi più economico considerare l'impiego del CD come supporto di distribuzione. I CD sono inoltre immuni dai campi magnetici e più durevoli dei floppy disk.

Sfortunatamente, fino a oggi i lettori di CD-ROM sono stati installati da una minoranza di utenti finali. La scarsa popolarità del CD-ROM è probabilmente dovuta al suo alto costo iniziale. Inoltre, gli utenti non li acquisteranno finché non ci sarà una quantità e qualità di software su CD-ROM tale da giustificare la spesa. Dal canto loro, i produttori non includono i lettori di CD-ROM nei loro sistemi di base per non incrementarne troppo il costo. Nonostante le difficoltà iniziali, piano piano il "fenomeno CD-ROM" sta comunque iniziando a diffondersi e nel giro di qualche anno è destinato sicuramente ad esplodere. Attualmente, la comunità Amiga ha due modi per saltare nel vagone del treno che sta correndo verso i CD-ROM: attraverso i prodotti ufficiali della Commodore oppure attraverso quelli di altre aziende.

I CD-ROM della Commodore

Nel momento in cui sto scrivendo, l'opzione della Commodore consiste nel CDTV e nel mini-CDTV per Amiga 500 battezzato A570. Il CDTV è un lettore di CD-ROM *stand-alone* che esteticamente

richiama i normali lettori di CD audio. Il CDTV può essere usato per leggere il software per Amiga registrato sui CD-ROM in formato CDTV, così come per riprodurre normali compact disc audio. È anche in grado di riprodurre i CD audio ai quali è stata aggiunta anche la codifica MIDI o grafica, ossia i formati CD+MIDI e CD+G, così come può leggere i CD registrati con lo standard ISO-9660. Bisogna però tenere conto del fatto che il CDTV non può eseguire il software per PC o Macintosh che spesso è fornito su dischi ISO-9660. Il CDTV è infatti pur sempre un computer Amiga, sul pannello posteriore dispone di tutte le porte di I/O standard e può essere trasformato in un computer a tutti gli effetti aggiungendo tastiera, lettore di floppy disk e mouse, tutti prodotti disponibili nel catalogo Commodore. Sebbene la Commodore abbia annunciato un controller SCSI per CDTV, questo dispositivo non si è ancora visto. Nel momento in cui scrivo, non c'è alcun modo per collegare un hard disk al CDTV o espanderne la memoria oltre il suo megabyte di chip RAM. Il CDTV può essere interfacciato a un computer Amiga attraverso la porta parallela e con il programma di pubblico dominio *ParNet*. Nella confezione del CDTV venduta qui negli USA è incluso il disco *Fred Fish collection on CD-ROM* della HyperMedia, che comprende anche il programma *ParNet*. Lo *street price* del CDTV, poi, è in continuo calo, anche perché contiene il chipset ECS dei vecchi Amiga (e non quello AGA dei nuovi A1200 e A4000).

L'A570 è un lettore di CD-ROM che si collega alla porta di espansione laterale dell'A500. Adesso che l'A500 non è più in produzione, il futuro di questa periferica è ovviamente segnato. L'A570 può essere espanso internamente con 2 MB di fast RAM. È prevista la possibilità di

collegare un'interfaccia SCSI opzionale, ma finora non si è ancora vista. Se acquistate un A570 e volete poi aggiungere un hard disk o un'espansione di memoria ulteriore, dovete quindi acquistare una periferica di una terza parte che si possa montare all'interno del computer.

Sulla base delle notizie divulgate all'ultima Conferenza dei programmatori, la possibilità d'integrare un CD-ROM su tutta la linea Amiga è una delle priorità principali per la Commodore. Però, ad oggi, la Commodore non ha ancora divulgato notizie su quando e a che prezzo e per quali macchine il CD-ROM farà la sua apparizione. È anche in fase avanzata la realizzazione del nuovo CDTV con i chip AGA di A1200 e A4000.

E allora procuriamoci un altro CD-ROM

Attualmente, impiegando i prodotti di terze parti è possibile aggiungere un lettore di CD-ROM a qualsiasi computer della linea Amiga. È necessario un drive CD-ROM, un controller compatibile e un driver software per gestire le comunicazioni e i trasferimenti di dati tra l'Amiga e il drive CD-ROM. La Xetec è stata la prima a offrire una soluzione CD-ROM completa per l'Amiga, ancora prima che fosse disponibile il CDTV. Il sistema della Xetec è quello che ho acquistato per il mio computer e posso dire di essere soddisfatto dei risultati. Il package originale della Xetec consisteva in un drive per CD-ROM Chinon CDS-431 e nel software di supporto per l'Amiga. Il software della Xetec supporta la maggior parte dei controller SCSI, compreso quello della Commodore. Attualmente, la Xetec fornisce il drive Chinon CDS-435. Il nuovo drive è più veloce del precedente e legge

anche i Photo CD della Kodak. Il prodotto della Xetec è disponibile sia per l'installazione interna (\$499), sia esterna (\$599). La soluzione esterna comprende un trasformatore e un contenitore per il drive. Il software della Xetec (\$50) può essere anche acquistato separatamente come driver *stand-alone*. Potete verificare direttamente con la Xetec quali drive CD-ROM possono essere utilizzati con il vostro controller SCSI e quali caratteristiche sono disponibili sul vostro Amiga. Il software della Xetec comprende *Fish & More Vol. 1 & 2*, la versione della Xetec della collezione Fred Fish e di altro software PD su due CD-ROM che arrivano fino al Fish numero 470.

Il software della Xetec va oltre la semplice lettura di CD-ROM sull'Amiga. È stato fatto un notevole sforzo per emulare l'operatività del CDTV. Sono inclusi diversi file di librerie per riprodurre la maggior parte delle funzioni del CDTV. Non manca anche una collezione di file di startup per la maggior parte dei titoli su CDTV. Anche se non è garantita una compatibilità al 100%, sul mio sistema sono riuscito a far funzionare tutti i titoli CDTV in cui mi sono imbattuto sino a oggi. Non sono invece supportati i formati CD+G e CD+MIDI, in quanto sono legati all'hardware del CDTV. Con

il drive CD-ROM possono essere riprodotti anche i CD audio. La Xetec include un programma player audio molto simpatico. Il sistema della Xetec offre inoltre l'accesso ai dati presenti sui CD-ROM registrati con il formato Macintosh Hierarchical File System.

Driver per CD-ROM sono disponibili anche presso altre due aziende. La Canadian Prototype Replicas offre il *CDROM-FS*, che negli Stati Uniti è distribuito dalla HyperMedia Concepts. Circa due anni fa, ho avuto modo di esaminare una versione preliminare di questo software. Da allora il programma è stato migliorato sostanzialmente. Più di recente, ha fatto il suo ingresso nell'arena di questi prodotti *AsimCDFS* della Asimware Innovations, che include anche un CD-ROM Fred Fish che arriva fino al disco 637. Ho dato un'occhiata alla versione 1.0 di *AsimCDFS*, ma non ho potuto valutarlo, in quanto il controller SCSI Xetec MiniFast che uso con il mio drive CD-ROM non era supportato. A parte la mancanza del supporto diretto del CDTV, entrambi i prodotti offrono comunque un supporto ai CD-ROM simile a quello del sistema della Xetec.

Tra i tre driver software, mi sento comunque di raccomandare quello della Xetec, dal momento che il software è arrivato al suo terzo anno di sviluppo ed è stato aggiornato parecchie volte.

Software per corrispondenza

C64 & AMIGA

Istruzioni in italiano!

Programmi di utilità
(per chi non vuole solo giocare)
Per ogni esigenza (decine e decine di titoli)
A basso costo (prezzi da videogame!)
Facili da usare (istruzioni chiare e in italiano!)

Grafica, musica, archiviazione/elaborazione
dati, videotitolazione, effetti audio/video,
programmazione, didattica, gestione file e
altro ancora!

Ecco alcuni esempi:

- Personal Budget (C64/Amiga), per tenere sotto controllo le finanze personali e gestire qualsiasi movimento di denaro (stipendi, spese, andamento di attività commerciali, situazione di conti correnti, ecc.). Fornisce riassunti scalari, bilanci e grafici, per formulare valutazioni globali e relative a particolari categorie di movimenti. Facilissimo da usare!
- Mondri 3D (Amiga), serie di programmi per creare con grande facilità immagini e animazioni tridimensionali di incredibile realismo (raytracing).
- Professional Graph (C64), per stampare grafici commerciali basati su dati specificati dall'utente. Prevede grafici bi e tridimensionali a barre semplici e multiple, a torta e a linee singole e multiple.
- Scroll-Up (C64/Amiga), genera testi di grandi dimensioni che scorrono sul video tipicamente in verticale ma anche in orizzontale, con l'ausilio di effetti speciali (comparsa, scomparsa, flash, ecc.). Ottimo per la titolazione delle vostre videocassette.

Per ricevere il catalogo gratuito (specificate Amiga o C64), inviate il vostro indirizzo a:

Studio Bitplane
casella postale 10942
20124 Milano

Cosa c'è di buono su CD-ROM

Prima d'investire la stessa cifra di un Amiga 1200 in un drive CD-ROM, vorrete probabilmente sapere a cosa vi servirà. Le uniche applicazioni native-Amiga attualmente disponibili su CD-ROM sono i titoli CDTV. Non conosco nessuno che ritenga che oggi esista anche un solo titolo su CDTV che valga l'acquisto di un lettore di CD-ROM. L'unica eccezione è la collezione Fred Fish su CD-ROM che per molti utenti può essere sufficiente motivo d'acquisto. L'ultimo CD uscito contiene più di 800 floppy della collezione, ed è disponibile anche in formato compresso per l'uso nell'ambito di BBS.

I CD-ROM su Amiga non sono però limitati a materiale specifico-Amiga. Dal momento che il file system ISO-9660 è stato adottato quasi universalmente, questo vuol dire che da un Amiga si può accedere praticamente a qualsiasi dato su CD-ROM. In particolare, sono molto popolari le collezioni di clip-art. L'unica richiesta è che i dati siano in un formato che possa essere gestito da qualche applicazione Amiga. Per esempio, *Art Department* della ASDG ha driver in grado di lavorare con file d'immagini bitmap nei

formati GIF, TIFF e altri. *Art Expression*, il nuovo pacchetto di grafica vettoriale della Soft-Logik, e *ProVector* con l'utility *PSImport* della Stylus, consentono d'importare ed editare file di clip-art EPS. Con gli editor di testi per Amiga si possono invece visionare e modificare i file di testo. Nei numeri passati, in questa stessa rubrica, vi ho parlato più volte di collezioni di clip-art e di dati su CD-ROM. Tenete presente che i programmi di visualizzazione e le utility di ricerca, disegnate per PC o Macintosh, non sono in grado di funzionare sotto AmigaDOS.

Il drive per CD-ROM può essere utilizzato anche dagli emulatori disponibili per Amiga. Per accedere al CD-ROM con l'emulatore Macintosh *A-Max* della ReadySoft, è necessaria una copia del driver Macintosh per drive CD-ROM. Il driver dev'essere installato nel System folder sulla partizione dell'hard disk usata da *A-Max*. È anche necessario creare una voce AMAX nella mountlist per il drive che va salvata nella directory DEVS: in modo che *A-Max* possa riconoscere il CD-ROM. Ho verificato che la voce della mountlist creata dal software della Xetec svolge anche questo compito.

Per poter accedere a un CD-ROM con un emulatore PC, è necessaria una Bridgeboard. Gli emulatori software disponibili per Amiga non sono infatti in grado di gestire l'interfaccia hardware necessaria per accedere al CD-ROM. Naturalmente, è necessario un Amiga con bus di espansione (come l'A2000) in grado di accogliere la Bridgeboard. Per completare il setup, il controller per CD-ROM va installato in uno degli slot Bridgeboard e il driver software va installato sul lato PC del sistema. Solitamente, i drive CD-ROM per PC sono venduti con l'interfaccia e il suo driver software. La Xetec offre l'interfaccia PC per il drive Chinon come opzione a un costo addizionale.

Il Photo CD della Kodak è l'ultimo vagone aggiunto al treno del CD-ROM. I Photo CD sono collezioni d'immagini fotografiche scansionate in alta risoluzione e memorizzate su CD-ROM. Per ottenere un Photo CD, s'impiega un normale rullino per macchine fotografiche. Se lo si porta a sviluppare in un centro autorizzato dalla Kodak, oltre al normale sviluppo, con circa 20 dollari le immagini vengono scansionate e trasferite su un disco Photo CD. I lettori di Photo CD costano circa \$400 e ricordano da vicino il CDTV e i normali lettori di CD audio. A differenza di qualche tempo fa, prevedendo un largo mercato, la maggior parte dei produttori di CD-ROM ha aggiunto la possibilità di leggere i Photo CD con i propri drive CD-ROM.

Ci sono due livelli di compatibilità con

il Photo CD: monosessione e multisessione. I drive monosessione possono leggere solo i Photo CD creati dai centri Kodak dopo lo sviluppo del film. A un Photo CD possono però essere aggiunte immagini addizionali successivamente alla codifica del primo rullino. I Photo CD con immagini aggiunte sono detti multisessione. In questo modo su un singolo Photo CD si possono registrare fino a 100 immagini. Ad oggi, la maggior parte dei lettori di CD-ROM compatibili con il Photo CD sono monosessione, anche se quasi tutti i produttori stanno sviluppandone delle versioni multisessione. Anche il Chinon CDS-435 è un lettore monosessione.

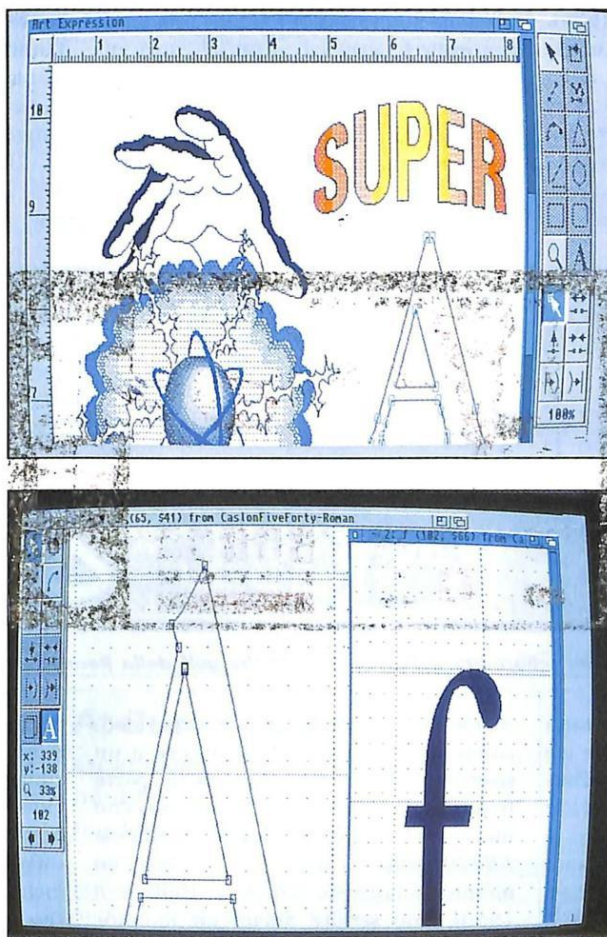
Da parte dell'utenza Amiga è stato espresso considerevole interesse nei confronti del Photo CD. Sfortunatamente, questo standard non è ancora disponibile per l'Amiga. Per poter usare i Photo CD sull'Amiga, bisogna iniziare acquistando un lettore di CD-ROM compatibile Photo CD. Poi, bisogna disporre di un driver CD-ROM in grado di estrarre i dati codificati in formato Photo CD. Un portavoce della Xetec mi ha comunicato che attualmente stanno verificando la possibilità d'includere le routine Photo CD nel loro file system per CD-ROM. Ma l'estrazione dei dati dal Photo CD rappresenta solo metà dell'opera. I dati codificati devono infatti essere convertiti in un formato d'immagine che possa essere visualizzato e manipolato dall'Amiga. Ho contattato la ASDG che mi ha comunicato che nonostante il grande interesse della comunità Amiga, la Kodak richiede il pagamento della somma di 250 mila dollari per consentire l'implementazione del supporto Photo CD all'interno di un programma come *Art Department*. Sfortunatamente, il mercato Amiga non è sufficientemente grande da garantire un ritorno da un investimento del genere su un pacchetto software.

Un portavoce della Kodak mi ha però riferito che l'interesse verso il Photo CD da parte degli utenti Amiga è tale che si sta valutando la possibilità di rendere disponibile questo standard per l'Amiga. Se anche voi siete interessati alla cosa, vi raccomando d'inviare una lettera in proposito alla Kodak: più richieste riceveranno, più ci sono possibilità che la cosa si concretizzi. Attualmente, la Kodak supporta il Photo CD sui sistemi MS-DOS,

Windows e Macintosh. Il software *Photo CD* della Kodak costa \$39.95 e viene fornito su CD-ROM. Il pacchetto consente di visualizzare le immagini e di esportarle in una varietà di formati di file diversi.

Art Expression

Questo programma ha avuto un gran numero di release. Ha iniziato con la versione 1.0. La copia per recensione che



Art Expression (sopra) e TypeSmith (sotto) della Soft-Logik

ho ricevuto era la versione 1.01, che funzionava bene con l'A1200, ma tavolta andava in crash sull'A2000. Inoltre, se si stampava attraverso il driver di *Preferences* con una Action Laser II della Epson, che è un clone della HP LaserJet IIP, il riquadro "in progress" finiva per creare problemi alla schermata presente in quel momento sull'A2000. Lo stampato risultava poi non centrato correttamente sulla pagina e leggermente fuori scala. La versione 1.02 di *Art Expression* ha corretto sia l'instabilità del programma, sia il problema con il display dell'A2000, ma i problemi di stampa sono rimasti. La versione 1.02 sembra anche che abbia bisogno di un tempo spropositato per

creare una pagina a 300 dpi sempre sulla stampante della Epson. Ho aspettato fino a 15 minuti, prima di spegnere tutto. Adesso, mi è arrivato *Art Expression 1.03*. Da una prima rapida verifica ho sperimentato che i problemi con i driver di *Preferences* non sono stati risolti. Mi è stato riferito che è già pronta la versione 1.04, ma che i problemi di stampa non sono ancora stati eliminati.

Questi problemi con *Preferences* sono davvero misteriosi. *PageStream*, il programma di desktop publishing della Soft-Logik, bypassa *Preferences*, impiegando driver di stampa proprietari per un'ampia varietà di stampanti. E questi driver con *PageStream* funzionano davvero bene. Apparentemente, *Art Expression* è stato realizzato da un consulente indipendente per conto della Soft-Logik al quale non è stata fornita la tecnologia dei driver di stampa sviluppata dall'azienda.

Una possibile circunnavigazione di questo problema con le stampanti laser è fattibile grazie a *Post*, il programma emulatore PostScript distribuibile gratuitamente, scritto da Adrian Aylward. *Post* è presente nella collezione Fred Fish. In *Post* è presente anche *PostLJ*, un'utilità di stampa PostScript per le stampanti LaserJet. Ho utilizzato *Post* con file PostScript scritti su disco da *Art Expression*. I risultati sono stati una stampa più veloce di quella di *Art Expression*, con in più i problemi di riscalatura e centratura eliminati. Forse alla Soft-Logik dovrebbero considerare l'inclusione di *PostLJ* nel package di *Art Expression*.

La Soft-Logik ha iniziato a commercializzare anche *TypeSmith 1.0*, un editor di fonti outline che può creare, modificare e salvare fonti outline. *TypeSmith* deriva da *FontDesigner*, un programma originariamente sviluppato dalla Relog e pubblicato in Germania dalla BSC. Quando ho parlato per la prima volta di *FontDesigner*, era stato introdotto nel Nordamerica dalla canadese Pre'Spect Technics. In seguito, i diritti di distribuzione sono stati acquistati dalla Soft-Logik, la quale dispone dei diritti per tutte le nazioni di lingua inglese. La BSC continua ad occuparsi del resto del mondo.

Sembra che la Soft-Logik abbia investito parecchi sforzi in *TypeSmith*. Ho avuto modo di vedere una versione preliminare

di questo pacchetto allo scorso Commodore Show di Toronto. La Soft-Logik ha rifatto completamente l'interfaccia utente e ha riscritto il manuale. Il formato di fonti nativo rimane il sistema di fonti outline della Soft-Logik. Il programma può anche importare ed esportare fonti Compugraphic e Adobe Type 1.

Novità dal fronte videogiochi

È uscito *The Lost treasures of Infocom II* della Activision. A differenza della prima versione di *Lost treasures*, questa collezione non verrà realizzata in formato Amiga. Per utilizzarla con il calcolatore della Commodore è quindi necessaria una scheda Bridgeboard o un emulatore come *CrossPC*. *Lost treasures II* comprende 11 dei classici della Infocom: *Border Zone*, *A Mind Forever Voyaging*, *Plundered Hearts*, *Bureaucracy*, *Cutthroats*, *Hollywood Hijinx*, *Seastalker*, *Sherlock the Riddle of the Crown Jewels*, *Wishbringer*, *Nord and Bert Couldn't Make Head or Tail of it* e *Trinity*. La confezione include le ristampe all'interno di un libro della documentazione inclusa nei package originali. Mancano invece gli hint book.

Tutto sommato, se siete degli appassionati dei prodotti della Infocom e disponete di un PC o di un emulatore, *Lost treasures II* è un buon affare.

La ReadySoft ha invece pubblicato *The Animation classics pack* per Amiga, Atari ST e PC. Si tratta di una compilation che ripropone tre giochi stile cartoon della ReadySoft: *Dragon's Lair II: Time Warp*, *Wrath of the Demon* e *Space ace*. Una nuova uscita è invece *War in the gulf*, gioco ambientato nel '95, anno in cui si ipotizza una nuova invasione del Kuwait da parte dell'Iraq. Il vostro compito è quello di controllare quattro gruppi di veicoli armati, guidati da mercenari veterani dell'operazione Desert Storm, che devono respingere l'invasione.

Alla scoperta del DCTV

Il sistema DCTV della Digital Creations per la visualizzazione e digitalizzazione non va confuso con il CDTV della Commodore, anche se era stata annunciata una versione del primo prodotto destinata al secondo. DCTV, che è l'acronimo di Digital Composite Television, è disponibile per l'Amiga già da qualche tempo. Recentemente, ne ho acquistato uno, e quella che segue è una descrizione frutto delle mie esperienze dirette.

Il DCTV (\$299) è un box nero delle

dimensioni di una scatola per carte da gioco un po' sovradimensionata. L'unità si collega alla porta video RGB dell'Amiga e dispone di un connettore pass-through per il collegamento del monitor RGB dell'Amiga. Un secondo cavo si collega alla porta parallela dell'Amiga. Il cavo per la porta parallela può essere lasciato scollegato quando non si vuole fare della videodigitalizzazione con il DCTV. Una coppia di jack standard RCA forniscono i collegamenti input e output per i segnali videocompositi. È disponibile sia una versione americana NTSC, sia una europea in PAL.

Per vedere l'immagine generata dal DCTV, il suo output videocomposito dev'essere collegato al corrispondente input di un monitor. I monitor RGB 1080/1084 della Commodore vanno benissimo per l'uso con il DCTV, dal

vare a 736 x 482 pixel. Le immagini possono essere create con il programma di disegno del DCTV che giudico potente e sofisticato quanto gli altri prodotti di paint con cui ho avuto modo di lavorare. La palette e il display a 24 bit sono in grado di emulare molti effetti, tra cui l'acquerello. Il pennello per l'acquerello ha anche la particolarità di asciugarsi a mano a mano che viene usato. Il DCTV è anche in grado di leggere e visualizzare qualsiasi immagine IFF a 24 bit. Le routine di input possono essere usate per centrare e riscalarle automaticamente l'immagine IFF affinché riempia l'intero display del DCTV. Ho utilizzato *Art Department Professional* della ASDG per convertire una selezione di file d'immagini TIFF e GIF da CD-ROM al formato IFF a 24 bit. *ADPro* include anche un operatore DCTV che è in grado di gestire direttamente il display del DCTV.

Il DCTV dispone di un videodigitalizzatore slow-scan incorporato. Il processo di digitalizzazione necessita di circa 10 secondi durante i quali l'immagine deve rimanere ferma. L'immagine viene digitalizzata utilizzando l'intera palette a 24 bit. Una videocamera o un videoregistratore con un buon fermo-immagine rappresentano una sorgente ottimale per il DCTV. Tenete presente che un display di 736 pixel per computer corrisponde a un display video di 368 linee. I

display video vengono classificati in termini di linee, mentre per quelli dei computer si parla di singoli pixel. I display video da 368 linee sono nelle possibilità della maggior parte delle videocamere. Il software di digitalizzazione del DCTV include controlli avanzati di image processing che possono essere impiegati per regolare luminosità, contrasto, colore e definizione dell'immagine. L'elaborazione dell'immagine si può applicare non solo alle immagini digitalizzate, ma a qualsiasi immagine caricata. Si possono gestire anche i file ANIM. Due anni fa, all'AmiExpo di New York, la IVS usava il DCTV per digitalizzare e riprodurre da hard disk, in 16 milioni di colori a 24 fotogrammi al secondo, uno spezzone tratto dal film *Ritorno al futuro 3*. La dimostrazione aveva lo scopo di far vedere quanto fosse veloce la scheda TrumpCard Professional della IVS nel leggere i dati da un hard disk SCSI. Ma faceva vedere anche ciò di cui è capace il DCTV. In definitiva, il DCTV è il dispositivo di visualizzazione a 24 bit, in 16 milioni di colori, meno costoso tra quelli disponibili per l'Amiga. Anche se



Una schermata tratta da *War in the gulf* della ReadySoft

momento che hanno un ingresso videocomposito sul pannello posteriore e un selettore RGB/composito nella parte frontale. Se l'Amiga è equipaggiata con una scheda de-interlacciatrice come la A2320 della Commodore, e con un monitor multisync (come il 1950 o il 1960), può servire anche un monitor separato con ingresso videocomposito. La maggior parte dei monitor multisync non hanno infatti alcun ingresso videocomposito. La Digital Creations offre anche un RGB Converter (\$199), che consente di vedere l'output del DCTV su un monitor RGB. L'RGB Converter gestisce anche genlock esterni.

Una volta che il DCTV è installato, l'Amiga continua a operare normalmente con il suo schermo del *Workbench* sul display RGB. Il software del DCTV è utilizzato per attivare il display videocomposito. Va tenuto presente che il DCTV non può essere utilizzato come un adattatore video *general purpose* per il *Workbench* o altri programmi, dal momento che genera un display in alta risoluzione, a 24 bit, con 16 milioni di colori. Le dimensioni dell'immagine in NTSC possono arri-

non dispone degli effetti speciali delle schede a 24 bit più avanzate, costa una frazione del prezzo di quest'ultime.

Deluxe Paint IV AGA

È disponibile il nuovo *Deluxe Paint IV* AGA della Electronic Arts. Tenete presente che richiede un *Workbench 2.04* o superiore e almeno 2 MB di memoria. Naturalmente, è anche necessario il chipset AGA per sfruttare le nuove modalità a 256 colori e il modo HAM8 a 262 mila colori. Se sono presenti il giusto *Kickstart* e sufficiente memoria, *Deluxe Paint IV*

funziona anche su macchine equipaggiate con il chipset originale o l'ECS. La confezione di *Deluxe Paint IV AGA 4.5* include lo stesso manuale di oltre 290 pagine della versione 4.0. Le differenze tra la versione AGA e la release precedente, sono descritte in un addendum di 15 pagine. Il programma viene fornito in formato compresso su tre dischi. Il disco *DeluxeVideo III Bonus Utilities*, presente nella versione precedente, non fa più parte del package. Il programma non è protetto, è però necessario introdurre nome dell'utente e numero di serie prima del run. All'avvio del programma, ci si

trova dinanzi al nuovo requester del formato di schermo che elenca tutti i modi di visualizzazione e rispettive caratteristiche disponibili sul proprio sistema. Lo stesso requester è utilizzato per cambiare i modi grafici durante l'esecuzione del programma. L'interfaccia utente è stata migliorata e si può accedere a tutti i requester direttamente da tastiera.

Deluxe Paint AGA gestisce anche le tavolette grafiche a pressione della TriMedia. Tra gli altri miglioramenti segnalo infine la gestione in lettura dei file IFF a 24 bit e il morphing a tutto schermo.

**Per ulteriori informazioni
contattare direttamente:**

Activision/Infocom
11440 San Vicente Drive,
Suite 300
Los Angeles, CA 90049
USA
(Tel. 001/310/8206131)

Asimware Innovations
101 Country Club Drive
Hamilton, Ontario
Canada L8K 5W4
(Tel. 001/416/5784916)

Canadian Prototype Replicas
PO Box 8
Breslau, Ontario
Canada N0B 1M0

Chinon America
660 Maple Ave.
Torrance, CA 90503,
USA
(Tel. 001/310/5330274)

Kodak
343 State Street
Rochester, NY 14650-0811,
USA

Electronic Arts
1450 Fashion Island Blvd.
San Mateo, CA 94404, USA
(Tel. 001/415/5722787)

HyperMedia Concepts
5200 Washington Ave. Suite 224
Racine, WI 53406, USA
(Tel. 001/414/6323766)

ReadySoft
30 Wertheim Court, Suite 2
Richmond Hill, Ontario
Canada L4B 1B9
(Tel. 001/416/7314175)

Soft-Logik Publishing
P.O. Box 510589
St. Louis, MO 63151-0589
USA
(Tel. 001/314/8948608)

TriMedia
60 E. Hintz Rd.
Wheeling, IL 60090, USA
(Tel. 001/708/5200730)

Xetec
2804 Arnold Road
Salina, KS 67401, USA
(Tel. 001/913/8270685)

Db-Line



0332/819104



Disponibile software per Amiga - Listino Gratuito

Oktagon - velocissimo controller SCSI-2 espandibile a 8mb, può gestire fino a 7 unità, compatibile con Amiga 4000, software GigaMem in omaggio.

At-Bus - controller At-Bus/IDE espandibile ad 8mb, può gestire 2 harddisk, compatibile con tutti gli harddisk At-Bus (anche da 2.5")

ZyXEL modem

Modem compatibile v32bis, v42bis con FAX (G3) e modulo VOICE. Standard proprietario fino a 19200bps (2300cps)

Oktagon 508 per Amiga 500 - OK
Oktagon 2008 prt Amiga 2000 - OK

At-Bus 508 per Amiga 500 - OK
At-Bus 2008 per Amiga 2000 - OK

ZyXEL U1496E - 16800 bps
ZyXEL U1496E+ - 19200 bps

Espansioni di memoria

CHIP RAM - SIMM

Schede grafiche

512k con clock per A500
1mb per A500(+)
2mb per Amiga 2000/3000/4000
PCMCIA da 2mb per A600/1200

SIMM da 2mb per A4000/MBX
SIMM da 4mb per A4000/MBX
ZIP PG per A3000/Oktagon/At-Bus
ZIP SC per A3000/Oktagon/At-Bus

Digital EditMaster - (non linear editing system)
AVideo/ColorMaster 12
AVideo/ColorMaster 24
Flicker Fixer per A2000 /A500

Disponibili lettori e dischi CD-ROM per Amiga (Fred Fish Collection), Ms-DOS (Windows, OS/2), Macintosh

Hard Disk SCSI

Hard Disk IDE 3.5"

Hard Disk IDE 2.5"

TruePaint - programma grafico a 24 bit per Amiga 1200/4000

VOXonFAX - listini, schede tecniche, novità, offerte speciali sono disponibili online per essere scaricate mezzo FAX. Richiedi il tuo codice d'accesso.

MBX MICROBOTICS
1200

Db-Line Srl - V.le Rimembranze 26/C - 21024 Biandronno (VA)
FAX: 0332/767244 - VOXonFAX: 0332/767360
BBS: 0332/706469-819044

LA GESTIONE DEI TESTI: ANALISI DELLE FUNZIONI

Continua la descrizione del modulo GestioneTesti.c facente parte di un programma di presentazione multimediale. In questa puntata vengono analizzate in dettaglio le funzioni principali di gestione dei testi

di Eugene P. Mortimore

Nell'articolo precedente abbiamo iniziato a esaminare in dettaglio il modulo *GestioneTesti.c*; ricordiamo che questo modulo si occupa della gestione di tutte le scene di tipo testuale facenti parte dell'applicazione multimediale oggetto di questa serie di articoli. In particolare, si è descritta la sequenza di chiamate a funzione del modulo, facendo anche uso di due schemi esplicativi, e si sono analizzate le principali strutture che entrano in gioco nella gestione dei testi. Abbiamo sottolineato come questi due tipi di analisi siano utili per la comprensione di un programma di una certa complessità.

Ora ci proponiamo di analizzare in dettaglio alcune delle funzioni principali del modulo *GestioneTesti.c*; la loro descrizione dovrebbe risultare facilmente comprensibile se si fa riferimento alle informazioni riportate nei precedenti articoli.

La funzione DisplayText

Ricordate che il punto di entrata iniziale del modulo *GestioneTesti.c* è rappresentato dalla funzione *DisplayText*. Tipicamente, il vostro programma chiamerà la funzione *DisplayText* ogni volta che l'utente selezionerà una gadget di una scena non testuale per visualizzarne una di tipo testuale. Quest'ultimo tipo di scena sarà costituita da un grande rettangolo bianco all'interno del quale compariranno tutte le parole della prima pagina di testo da visualizzare. Normalmente, la maggior parte delle parole del testo appariranno in nero su sfondo bianco; le parole chiave saranno invece evidenziate in un colore differente. Inoltre, le parole chiave su cui l'utente muove il puntatore del mouse verranno evidenziate utilizzando un terzo colore ancora.

Dal listato pubblicato negli articoli precedenti, potete vedere che la funzione *DisplayText* inizia impostando il gadget del puntatore allo stato *DO_WAIT* e impostando la variabile globale *CurrentPageNum* al valore "0", cioè al numero della prima pagina di testo. Notate che la nostra definizione della funzione *SetPointers* supporta l'uso sia del mouse che dei tasti cursore della tastiera o del telecomando del CDTV.

Nel nostro caso, in cui vogliamo impostare il gadget del puntatore allo stato *DO_WAIT*, la funzione *SetPointers* dapprima controlla se si sta utilizzando il mouse *Intuition*. Se così è, la funzione *SetPointers* chiama la funzione della "intui-

tion.library" *ModifyIDCMP*, per comunicare a *Intuition* d'ignorare gli eventi dovuti al tasto sinistro del mouse. Il programma chiama poi la funzione *ReportMouse* di *Intuition* per non farsi più segnalare gli eventi relativi al mouse, chiama poi la funzione di *Intuition* *SetPointers* per impostare l'immagine del puntatore a *WaitPointer* e infine la funzione *ChangeSprite* della "graphics.library" per impostare l'immagine grafica associata allo sprite di programma a un'immagine trasparente, in modo che l'immagine della freccetta non appaia sullo schermo dell'utente. Potete verificare dal listato che, anche nel caso non si stia utilizzando il mouse, il programma esegue una serie equivalente di azioni.

A questo punto, se la scena corrente non è di tipo *Help*, la variabile globale *CurrentLanguage* viene impostata al valore della variabile globale *LocalLanguage*. Ricordate che la variabile globale *LocalLanguage* è normalmente impostata all'inizializzazione del programma e si vuole che la prima pagina di testo appaia inizialmente nella lingua impostata mediante questa variabile. Una volta che la pagina di testo è sullo schermo, l'utente può selezionare un gadget di selezione lingua per visualizzare la stessa pagina di testo in un'altra lingua.

La funzione *DisplayText* imposta a questo punto le variabili globali *TextPageWidth* e *TextPageHeight*: sono le variabili che determinano con la precisione del pixel come il testo viene visualizzato sullo schermo ad alta risoluzione. Queste due variabili dovrebbero permettervi di stabilire un bordo bianco intorno al testo; verranno infatti usate più tardi nella funzione *DisplayText* per determinare il formato di tutte le successive pagine di testo all'interno dell'appropriata area di schermo.

Il compito successivo della funzione *DisplayText* è d'impostare i valori delle coordinate in pixel dei gadget di paginazione. Si ricordi che i gadget *PageUp* e *PageDn* normalmente appariranno in due diverse posizioni a seconda che ci si trovi su una generica scena di tipo testuale o su una scena testuale di *Help*. Si noti anche che la funzione *SetPageGadgets* accetta soltanto un argomento, ovvero la scene list appropriata.

Una volta che le coordinate dei gadget di paginazione sono state impostate, la funzione *DisplayText* chiama la funzione *AllocPagesText*. Questa funzione alloca la memoria necessaria per contenere le bitmap che il *Blitter* userà per visualizzare velocemente sullo schermo ciascuna nuova pagina di testo,

quando l'utente seleziona un gadget di paginazione. Esaminando il listato di questa funzione, potete vedere che il risultato finale della sua esecuzione è quello di chiamare la funzione `AllocPage` due volte, una per la pagina di testo stessa e un'altra per il testo di definizione delle parole chiave.

Una volta allocata l'appropriata quantità di memoria per le bitmap, la funzione `DisplayText` inizializza le fonti con le quali il testo verrà visualizzato, chiamando la funzione `InitializeFonts`. Si noti che tale funzione imposta inizialmente la fonte a `textFontTimes18`, che va bene per il caso in cui il programma venga eseguito su un Amiga dotato di drive per CD-ROM A570 e monitor ad alta risoluzione. Tuttavia, se il programma viene eseguito su una macchina CDTV NTSC collegata a una normale TV, la fonte viene dapprima cambiata in una più grande, `textFontTimes24`, e se la macchina è di tipo PAL la fonte viene cambiata in una ancora più grande, `textFontTimes30`.

Una volta che l'impostazione delle fonti è terminata, la funzione `InitializeFonts` chiama la funzione `MySetFont` per impostare la fonte in `frontWindowRastPort` e poi chiama due volte ancora la funzione `MySetFont` per impostare la fonte sia per la `RastPort` della pagina di testo che per la `RastPort` della definizione delle parole chiave.

Si noti che la funzione `MySetFont` imposta le variabili `TextLengthSpace`, `YIncrement` e `BaseLine`, che saranno usate per spaziare opportunamente il testo in tutte le bitmap relative (quella associata a `frontWindowRastPort`, quella della pagina di testo e quella della definizione delle parole chiave). Terminati tutti questi preliminari, la funzione `DisplayText` può ora ottenere il nuovo blocco di testo leggendolo dal CD-ROM, metterlo in un buffer di testo, dividerlo nelle singole parole e visualizzarne la prima pagina di testo con tutte le parole chiave opportunamente evidenziate. Questo è lo scopo della funzione `CreateWordList`.

La funzione `CreateWordList`

Il modulo *GestioneTesti.c* chiama la funzione `CreateWordList` ogni volta che l'utente seleziona un gadget per visualizzare una nuova pagina di testo sullo schermo. Se il gadget selezionato dall'utente serviva a visualizzare una nuova scena di tipo testuale, tale funzione viene chiamata attraverso la funzione `DisplayText` e la pagina di testo visualizzata è la prima; se invece il gadget selezionato è `PageUp`, `PageDn` o un gadget di selezione lingua, viene visualizzata la corrispondente pagina di testo della stessa scena. Quindi, la funzione `CreateWordList` può essere chiamata, oltre che dalla funzione `DisplayText`, da altri punti del programma, come si può vedere dalle figure e dal listato pubblicato nei precedenti articoli. In particolare, potete trovare chiamate alla funzione `CreateWordList` nella funzione `ProcessTextWindowLeftMouseButtons`; queste chiamate vengono effettuate quando l'utente seleziona un gadget di paginazione o un gadget di selezione lingua, mentre è visualizzata una scena di tipo testuale.

La funzione `CreateWordList` inizia visualizzando un messaggio per avvertire l'utente di attendere che il programma termini qualche operazione interna. Queste operazioni riguardano la lettura dell'opportuno blocco di testo dal disco CD-ROM, la divisione del blocco in singole parole, il calcolo di tutte le coordinate di schermo per il testo e infine la visualizzazione del testo sullo schermo. Queste operazioni possono durare parecchi secondi; il tempo reale dipende dalle dimensioni del blocco di testo, cioè dal numero di parole in esso presenti, e dalla presenza o meno di parole chiave da evidenziare. Su un Amiga 3000 con una CPU 68030 il programma può analizzare diciotto pagine di testo e visualizzare la prima entro tre secondi. Su una macchina CDTV con un più lento 68000 queste operazioni durano

almeno il doppio del tempo.

Potete vedere che la funzione `ShowPleaseWait` dapprima determina se la macchina è PAL o NTSC e poi chiama la funzione `MyRectFill` per disegnare un rettangolo di dimensioni opportune nell'area di testo, poi si posiziona più o meno al centro di tale rettangolo e scrive un messaggio per avvertire l'utente di attendere. Probabilmente, vorrete fare in modo che il messaggio effettivamente utilizzato dipenda dalla lingua locale: per fare questo, basta controllare il valore della variabile `LocalLanguage` e usare la funzione `Text` della "graphics.library" per visualizzare la frase appropriata. Poi la funzione `CreateWordList` chiama la funzione `BuildWordList` per analizzare il nuovo blocco di testo. Notate che la funzione `BuildWordList` ha un solo argomento, il nome del blocco di testo, mentre la posizione di tale blocco è determinata sfruttando come indice l'array `FileDef` basandosi sul valore corrente della variabile globale `SceneIndexCurrent`.

Ora dobbiamo distinguere due casi: scene di tipo Help e normali scene testuali. In quest'ultimo caso possiamo usare la variabile globale `SceneIndexCurrent` direttamente come viene passata dal modulo *GestioneMessaggi.c* al modulo *GestioneTesti.c*. L'array `FileDef` è infatti impostato in modo da funzionare correttamente in questo caso. D'altra parte se la scena corrente è di tipo Help dobbiamo modificare localmente e poi ripristinare, all'interno della funzione `CreateWordList` stessa, la variabile globale `SceneIndexCurrent` per puntare alla giusta posizione nell'array `FileDef`, dove sono memorizzate le informazioni relative al testo di Help. Per fare questo, usiamo la variabile `SceneIndexTemp`, locale alla funzione `CreateWordList`. Ora, se la scena testuale è di Help impostiamo la variabile locale `SceneIndexTemp` al valore corrente della variabile globale `SceneIndexCurrent` e impostiamo quest'ultima variabile al valore della variabile globale `HelpIndexCurrent`. Si ricordi che la variabile `HelpIndexCurrent` è stata impostata nel modulo *GestioneMessaggi.c* quando si sta per visualizzare una scena di tipo Help. Questa è ancora una volta una delle caratteristiche del programma che dovrete adattare alla vostra strategia di visualizzazione delle scene.

A questo punto, con la variabile `SceneIndexCurrent` eventualmente modificata a seconda del tipo di scena testuale corrente, possiamo chiamare la funzione `BuildWordList` e l'indice `FileDef` ci farà puntare alla giusta posizione del blocco di testo nell'opportuno file. Osservate che se la scena era di tipo Help, alla fine della funzione `CreateWordList` vi è il ripristino della variabile `SceneIndexCurrent`, in modo che all'uscita da questa funzione tale variabile ritorni al suo valore originale.

Questa procedura è un esempio di modifica e successivo ripristino del valore di una variabile globale localmente a una funzione, che permette di ridurre la dimensione del codice. L'alternativa a questo metodo sarebbe stata quella di creare due array `FileDef` diversi per le scene testuali normali e di Help, ma questo schema non sarebbe stato altrettanto economico sia per le dimensioni del programma sia per la sua esecuzione.

Una volta che la lista di parole è costruita, la funzione `CreateWordList` chiama la funzione `SetWordCoordinates` per impostare le coordinate in pixel di tutte le parole nel blocco di testo appena analizzato dalla funzione `BuildWordList`. Ancora una volta, queste operazioni occupano molto la CPU e possono durare diversi secondi a seconda della CPU usata.

Una volta che la funzione `SetWordCoordinates` ritorna, la funzione `CreateWordList` può chiamare la funzione `SetHotWords`. Più precisamente, se la variabile globale `ShowHotWords` è impostata a `TRUE` nel modulo *GestioneMessaggi.c* per questo blocco di testo, viene chiamata la funzione `SetHotWords` per identificare le parole chiave ed etichettarle come tali impostando gli opportuni bit.

Ora tutti i preliminari sono completati e la funzione `CreateWordList` può chiamare la funzione `DrawWordList` per disegnare effettivamente tutte le parole del blocco di testo. Quando la funzione `DrawWordList` ritorna, l'utente vedrà una nuova pagina di testo visualizzata sullo schermo e se la variabile `ShowHotWords` era impostata a `TRUE` tutte le parole chiave appariranno all'interno di un piccolo rettangolo il cui colore è diverso dal bianco di sfondo delle altre parole.

La funzione `BuildWordList`

La funzione `BuildWordList` è una delle funzioni più importanti del modulo *GestioneTesti.c*. Essa ha quattro scopi:

- liberare la memoria dinamicamente allocata da precedenti chiamate alla funzione `BuildWordList` stessa. Questa memoria è costituita dagli eventualmente molti piccoli buffer necessari per contenere una struttura `Word` per ciascuna parola del blocco di testo;
- posizionarsi opportunamente all'interno del disco CD-ROM e leggere il nuovo blocco di testo, ponendolo in un ampio buffer di memoria allocato dinamicamente;
- analizzare l'intero blocco di testo, dividendolo in un insieme di parole per ciascuna delle quali viene allocata dinamicamente una struttura `Word`. L'insieme delle strutture `Word` va a costituire una lista che rappresenta tutte le parole di una o più pagine di testo;
- liberare il buffer precedentemente allocato per l'intero blocco di testo.

Per capire come la funzione `BuildWordList` riesca a gestire più lingue locali, bisogna studiare dapprima la funzione `FreeWordListMem`, che è la prima funzione chiamata da `BuildWordList`. Si può vedere che la funzione `FreeWordListMem` chiama la funzione `FreeMem` della "exec.library" per ciascuna parola della precedente lista. Questo viene fatto con un ciclo "for" che parte dalla parola numero "0" e finisce alla parola indicata dalla variabile `ActualNumWords`. Questa variabile è inizializzata a "0" ed è aggiornata contando le parole contenute in ciascun nuovo blocco di testo letto dalla funzione `BuildWordList`. Quindi, se una lista di parole è vuota, questo ciclo "for" non ha iterazioni; è quello che succede quando la funzione `BuildWordList` viene chiamata per la prima volta. Si noti anche che la funzione `FreeWordListMem` non chiama la funzione `FreeMem` a meno che il puntatore `wordlist` sia diverso da `NULL`. Questo è il modo corretto per gestire l'allocazione e la disallocazione dinamica delle strutture `Word`.

Quando la funzione `FreeWordListMem` ritorna, la funzione `BuildWordList` è pronta per chiamare la funzione `GetTextBuffer`. È dentro questa funzione che viene allocata la memoria per il nuovo blocco di testo che viene letto dal disco. Le principali funzioni chiamate da `GetTextBuffer` sono le funzioni `AllocMem` di "exec.library" e le funzioni `Seek` e `Read` di "dos.library". È stata progettata una funzione `GetTextBuffer` abbastanza generale da potersi posizionare correttamente sulla giusta linea di testo di qualsiasi file specifico della lingua (un file per l'inglese, uno per l'italiano, uno per il francese, eccetera) o anche per posizionarsi correttamente all'interno di un file di definizione delle parole chiave.

Quello che serve per poter chiamare la funzione `AllocMem` è determinare il giusto file, la giusta dimensione del buffer di testo e la linea d'inizio e di fine all'interno del file (ogni linea è lunga "100" caratteri), in modo da poterle leggere nella nuova memoria allocata.

Si ricordi che il file da leggere dipende dal valore della variabile globale `CurrentLanguage`. Normalmente, questa variabile avrà lo stesso valore della variabile `LocalLanguage`, ma

se l'utente ha selezionato un gadget di selezione lingua, il valore della variabile `CurrentLanguage` viene modificato e la funzione `GetTextBuffer` preparerà le sue variabili locali per riflettere la nuova situazione. Potete vedere che il codice fa l'assunzione iniziale che il file corrente sia quello contenente il testo inglese, quindi le variabili `fileHandle`, `LineNumBgn` e `LineNumEnd` vengono impostate secondo questa assunzione. Tuttavia, lo "switch" sulla variabile globale `CurrentLanguage` risulterà in differenti valori per queste variabili se `CurrentLanguage` corrisponde per esempio all'italiano o al francese. Alla fine, la variabile `TextBufferSize` verrà calcolata basandosi sul valore corrente delle variabili `LineNumBgn` e `LineNumEnd` e utilizzando il valore `LINE_BUFFER_SIZE` pari a "100" byte.

Tutta la descrizione fatta sopra vale nel caso che la variabile `GetHotWordText` sia `FALSE`, il caso di default per la funzione `GetTextBuffer`. Se tuttavia il valore di questa variabile è `TRUE`, perché l'utente ha selezionato una parola chiave per visualizzarne la sua definizione, il file corretto da leggere è il file di definizione delle parole chiave. In questo caso la funzione `GetTextBuffer` reimposta alcune variabili, come `fileHandle` e `TextBufferSize`. Ricordate che per ogni parola chiave nel file di definizione sono contenute sette linee di testo di definizione; quindi la variabile `TextBufferSize` è ricalcolata basandosi su queste informazioni. Inoltre, il file di definizione delle parole chiave è uno solo per tutte le lingue usate dal programma e nel nostro esempio contiene tutte le definizioni in inglese per prime, seguite da quelle francesi e da quelle tedesche (nell'ordine). Tenendo conto che ogni definizione è lunga sette linee, la variabile `LineOffset` viene impostata in modo da corrispondere alla giusta posizione all'interno del file di definizione delle parole chiave. Il corretto valore di `LineOffset` dipenderà da come disporrete le definizioni per le varie lingue nel vostro file.

Per la definizione di una parola chiave, l'ultima cosa che rimane da fare è reimpostare la variabile `LineNumBgn`. Potete vedere dal listato che questa variabile viene calcolata utilizzando la variabile `LineOffset` insieme con il valore della variabile globale `HotWordIndex`. Ancora una volta, il calcolo della variabile `LineNumBgn` tiene conto del fatto che le definizioni delle parole chiave per una certa lingua partono da una specifica linea all'interno di un unico file e che ciascuna definizione è lunga sette linee.

Una volta impostate tutte queste variabili e identificato il giusto file, si può riempire il buffer di testo con i caratteri contenuti nelle linee lette dal file e mettere un carattere `NULL` alla fine del buffer. Quindi, quando la funzione `GetTextBuffer` ritorna alla funzione `BuildWordList`, si può calcolare l'effettiva dimensione del buffer di testo usando la funzione `strlen` e impostando così il valore della variabile `TextStringLength`. Ora può iniziare la fase di analisi del testo.

L'analisi del buffer di testo

Lo scopo dell'analisi del buffer di testo appena riempito è di separarne i caratteri in un insieme di parole distinte, in modo tale che sia un processo molto efficiente e che tenga comunque conto delle peculiarità di ciascuna lingua, nel nostro esempio inglese, francese e tedesco. Sapendo che questo è lo scopo, si può formulare una strategia di analisi che cerchi tutti gli spazi e che termini una parola quando trova uno spazio. Una volta trovata una parola, questa viene terminata con un carattere `NULL`; è il momento giusto per allocare dinamicamente il piccolo blocco di memoria dove inserire la struttura `Word` che identifica le caratteristiche principali di questa nuova parola, il che rende la gestione più efficiente.

La funzione `BuildWordList`, inoltre, deve marcare certe parole con il bit `SKIPWORD`. Lo scopo di questo bit è d'istruire

le successive funzioni perché saltino quella parola quando elaboreranno l'intera lista, che può essere anche molto grande. Questa caratteristica è importante specialmente quando la variabile globale ShowHotWords è impostata a TRUE. In quel caso, si vuole che l'identificazione delle parole chiave sia la più efficiente possibile. Quindi, per esempio, se la lingua corrente è l'inglese, e sapete che tutte le parole chiave inglesi sono lunghe meno di 13 caratteri, potete marcare tutte le parole inglesi più lunghe di 13 caratteri con il bit SKIPWORD. Allo stesso modo, se la lingua corrente è il tedesco, e sapete che tutte le parole chiave tedesche sono meno lunghe di 17 caratteri, potete marcare tutte le parole tedesche più lunghe di 17 caratteri con il bit SKIPWORD. Un comportamento simile può essere usato anche per la lingua francese o italiana. Inoltre, se le parole sono maiuscole e iniziano con la parentesi tonda aperta "(", possono essere marcate con il bit SKIPWORD. Uno studio accurato della funzione BuildWordList vi darà un'idea più chiara delle considerazioni qui fatte. Per esempio, se volete aggiungere l'opzione della lingua italiana alla funzione BuildWordList, dovrete per prima cosa compilare la lista delle parole chiave italiane e studiarne le caratteristiche. Può darsi che non troviate parole chiave più lunghe di "10" caratteri. Quindi, potrete impostare il bit SKIPWORD per tutte le parole italiane più lunghe di "10" caratteri. Potete poi trovare altre caratteristiche delle parole italiane per rendere più efficiente la gestione delle parole chiave.

Notate dal listato che le istruzioni al di sotto della label CONT2 servono per marcare le parole con bit addizionali. Per esempio, se una parola finisce con una ",", bisogna marcarla con due bit, il bit COMMA e il bit HASEND. Se la parola finisce con "." bisogna marcarla con il bit PERIOD e il bit HASEND. Lo stesso si può dire per altri tipi di finali di parola, come si può vedere dal codice.

Inoltre, se la finale di parola è "s", se la variabile globale CurrentLanguage è impostata a Inglese, e se la parola inizia con lettera minuscola, bisogna marcarla con il bit HASS (che significa appunto "ha la s") e il bit HASEND. Tutto ciò, per prendere in considerazione il modo in cui le parole inglesi formano il plurale; si assume inoltre che altre lingue non trattino le parole terminanti in "s" come casi speciali.

Le rimanenti istruzioni della funzione BuildWordList impostano un nuovo valore per la variabile ActualNumWords e liberano la memoria precedentemente allocata per contenere l'intero blocco di testo.

La funzione SetWordCoordinates

Potete vedere dagli schemi pubblicati nel precedente articolo che la funzione SetWordCoordinates è alla fine della sequenza di chiamate a funzioni ed è chiamata solo dalla funzione CreateWordList. Lo scopo della funzione SetWordCoordinates è d'impostare le coordinate di schermo per tutte le parole e per tutte le pagine del blocco corrente di testo. Tale funzione fa la stessa cosa per tutte le parole del testo di definizione di una parola chiave, ma in questo caso il testo è limitato a una sola pagina. L'obiettivo in questo caso è di calcolare le coordinate delle parole in modo che siano all'interno del rettangolo bianco sullo schermo, opportunamente distanziate dai bordi del rettangolo. Inoltre, le parole devono essere visualizzate in un formato tale da permettere all'utente di leggere il testo sullo schermo facilmente e in modo scorrevole.

Come potete vedere, la funzione SetWordCoordinates esegue un ciclo "for" su tutte le parole della lista corrente. Abbiamo appena visto che questa lista è stata costruita dalla funzione BuildWordList e che ogni parola è rappresentata da un'istanza della struttura Word. L'ultima parola della lista ha lunghezza

"0" e questo permette di terminare il ciclo "for". La funzione SetWordCoordinates deve impostare i valori corretti per le variabili TopLeftX, TopLeftY, BotRightX e BotRightY per ciascuna struttura Word. Inoltre, deve impostare il valore di PageNum nella struttura Word e stabilire il massimo numero di pagine per il blocco corrente di testo. Tenete conto che tutti questi valori dipenderanno necessariamente dalle caratteristiche della fonte in uso, come impostate dalla funzione InitializeFonts.

Per fare questo, la funzione SetWordCoordinates lavora con tre variabili locali: XCoor, coordinata X della parola, YCoor, coordinata Y della parola, e XIncrement, valore dell'incremento X pari a uno spazio più una parola nella fonte corrente. Inoltre, la funzione SetWordCoordinates accede a un certo numero di variabili globali precedentemente impostate: TextOffsetX, TextOffsetY, TextPageWidth, TextPageHeight, YIncrement e TextLengthSpace. Ricordate che le ultime due variabili dipendono dalle caratteristiche della fonte e che vengono impostate dalla funzione InitializeFonts.

La prima parte del ciclo "for" stabilisce quali parole sono caratteri speciali di formattazione, cioè caratteri inseriti nel blocco originale di testo solo per impostare il formato del testo che viene visualizzato sullo schermo. Naturalmente, il più semplice carattere di formattazione è lo spazio " ". Ricordate che, indipendentemente dalla lingua usata, la funzione BuildWordList usa gli spazi per identificare le singole parole all'interno del blocco di testo. Il carattere "spazio" viene visualizzato come una piccola porzione di schermo bianco, nel senso che lo spazio è il carattere che separa le singole parole sullo schermo.

Quando il programma deve posizionare il testo sullo schermo deve fare le seguenti cose:

- stampare una o più linee vuote per separare i paragrafi sullo schermo. Normalmente, manterrete i vostri paragrafi abbastanza corti per facilitare la lettura del testo da parte dell'utente; dovrete tenere a mente questo particolare quando costruite il vostro file di testo. Nel nostro esempio possiamo usare il carattere "@" come delimitatore di paragrafo. Potete vedere che in questo caso la variabile YCoor viene incrementata del doppio del valore della variabile YIncrement, cioè due volte l'altezza della fonte corrente;
- fornire un modo per forzare un a capo sullo schermo senza introdurre linee vuote nel testo. Questo è utile quando dovete visualizzare un gruppo di linee in una tavola o in una lista: in questo caso vorrete che ogni nuova linea di testo venga visualizzata direttamente sotto la linea precedente, senza interporre righe vuote. Nel nostro esempio possiamo usare il carattere "^" per questo scopo. Potete vedere che in questo caso la variabile YCoor viene incrementata del valore della variabile YIncrement, cioè esattamente dell'altezza della fonte corrente;
- fornire un modo d'indentare una linea con un numero fisso di spazi. Anche questo è utile quando dovete creare una lista sullo schermo con un titolo principale seguito da una serie di sottotitoli indentati. Nel nostro esempio possiamo usare il carattere "_" per questo scopo. Potete vedere che in questo caso la variabile XCoor è incrementata di un valore pari a quattro volte la variabile TextLengthSpace, cioè esattamente quattro volte la larghezza di uno spazio nella fonte corrente.

Si noti che la scelta da noi fatta per i caratteri di formattazione assume che questi caratteri non appaiano normalmente nel testo. Se tuttavia questa assunzione non è vera nel vostro caso, potete fare una diversa scelta di caratteri di formattazione. La spiegazione dettagliata delle funzioni fatta in questo articolo vi permetterà di studiare per conto vostro le altre funzioni del modulo *GestioneTesti.c*.

CORSO DI ASSEMBLY: OTTAVA PUNTATA

LE STRUTTURE DI DATI E L'INPUT/OUTPUT

Gli argomenti principali di questa puntata sono la creazione e la gestione delle strutture di dati e l'I/O in linguaggio Assembly

di Giovanni Zito

Nelle puntate precedenti ci siamo occupati delle principali istruzioni del 68000. In questa puntata ci occuperemo di due aspetti pratici: vedremo come creare e utilizzare alcuni oggetti tipici dei linguaggi di alto livello (variabili, vettori, strutture, stack frame...), e forniremo le direttive necessarie per un corretto utilizzo delle librerie di sistema da linguaggio Assembly. Vedremo inoltre come sia possibile l'interazione con il mondo esterno grazie alle funzioni della libreria DOS.

È noto che in un programma non compaiono solo istruzioni, ma anche dati. Infatti, qual è lo scopo dei programmi se non quello di manipolare delle informazioni? Queste ultime normalmente si presentano in diversi modi. Sappiamo che la più piccola informazione che un elaboratore possa elaborare è il bit. Seguono le variabili semplici (byte, word, long...) e quelle composte (vettori, strutture), fino ad arrivare ai file, i quali a loro volta non sono altro che insiemi di dati. Se ci pensiamo un attimo, non esiste nessun programma il cui fine ultimo non sia il trattamento dei dati, siano essi cifre, testi, immagini, suoni, animazioni o altro ancora.

Le variabili e gli array

Di certo tutti voi conoscete le variabili. In linguaggio C, per esempio, si può definire una variabile specificandone il tipo e assegnandole un valore. La seguente riga di programma definisce la variabile `cont` come unsigned long e le assegna il valore 200:

```
ULONG cont = 200;
```

Ora è importante capire che in realtà le variabili non sono altro che riferimenti a locazioni di memoria. Da questo punto di vista si può parlare di variabili anche in linguaggio Assembly. Per definire una variabile in Assembly scriveremo:

```
CONT: DC.L 200
```

Come già sappiamo, la direttiva `DC` ha l'importante funzione di definire dei dati all'interno del programma. In questo esempio riserviamo quattro byte di memoria e vi scriviamo la parola lunga `$000000C8` (200). Successivamente potremo accedere a questa "variabile" scrivendo:

```
MOVE.L CONT,D0
```

Questa istruzione trasferisce in `D0` la long word contenuta all'indirizzo `CONT` (modo di indirizzamento assoluto). L'istruzione:

```
MOVE.L #CONT,A0
```

caricherà in `A0` il valore rappresentato dall'etichetta `CONT`, cioè in definitiva l'indirizzo nel quale si trova la nostra long word. Si otterrebbe lo stesso effetto scrivendo:

```
LEA CONT,A0
```

In linguaggio C avremmo scritto `&cont` ("`&`" è un operatore che applicato a una variabile ne restituisce l'indirizzo), in AMOS `Varptr(CONT)`... Nel caso desiderassimo modificare il valore di questa variabile, per esempio in 1, potremmo scrivere:

```
MOVE.L #1,CONT ; copia 1 all'indirizzo CONT
```

In AMOS avremmo scritto:

```
CONT = 1
```

o anche:

```
LOKE Varptr(CONT),1
```

Quest'ultima istruzione concettualmente è molto simile alla mnemonica Assembly `MOVE.L #1,CONT`.

E dalle variabili semplici passiamo ora ai vettori. Mentre le variabili semplici consentono di associare un nome simbolico a una singola informazione, grazie ai vettori (chiamati anche array) è possibile riunire sotto un unico nome simbolico (il nome dell'array) insiemi di dati dello stesso tipo. Dal momento che tutti gli elementi dell'array hanno lo stesso nome, ciascun elemento viene univocamente identificato mediante un indice (al primo elemento viene associato l'indice 0, al secondo l'indice 1 e così via). In linguaggio C possiamo definire un array contenente 32 elementi di tipo unsigned word scrivendo:

```
UWORD palette[32];
```

Con questa istruzione il compilatore allocherà memoria per 32 variabili di tipo unsigned word (si veda la Figura 1):

```
palette[0]  
palette[1]  
palette[2]  
...  
palette[31]
```

Il compilatore C memorizza quindi l'indirizzo del primo elemento dell'array (`&palette[0]`) nel nome stesso dell'array, che perciò può essere utilizzato come puntatore a una word (`UWORD *`). In altri termini, la variabile `palette` viene a contenere l'indirizzo dell'elemento `palette[0]`. Cerchiamo ora di

guardare un array dal punto di vista della macchina. Si tratta di una zona di memoria (buffer) la cui dimensione viene stabilita al momento della dichiarazione dell'array. Nell'esempio precedente il buffer ha una dimensione di 64 byte (32 elementi, ciascuno ampio 2 byte). Come creare un array in Assembly?

```
palette: DCB.B 64,0
```

Ricordiamo che la direttiva DCB (in alcuni assembler BLK) definisce un blocco di dati all'interno del programma. Il primo operando (64) indica le dimensioni del buffer, mentre il secondo operando (0) rappresenta il valore con cui inizializzare l'intero blocco di memoria. Avremmo potuto scrivere anche:

```
palette: DCB.W 32,0
```

ovvero:

```
palette: DCB.L 16,0
```

L'etichetta palette rappresenta l'indirizzo del nostro buffer/array, o, che è lo stesso, l'indirizzo del primo elemento dell'array (palette[0]). Posto che quanto detto sia ben chiaro, il passo successivo consiste nell'apprendere in che modo sia possibile indirizzare un certo elemento dell'array, per esempio palette[4]. Il problema è che in Assembly non possiamo fare uso di nomi di variabili, né tanto meno d'indici, perché l'Assembly lavora soltanto con gli indirizzi. Comunque, dal momento che ogni elemento dell'array consta di 2 byte, possiamo facilmente calcolare l'indirizzo dell'elemento palette[4], come si vede dal seguente schema:

Assembly	Linguaggio C	AMOS
palette	&(palette[0])	Varptr(PALETTE(0))
palette+2	&(palette[1])	Varptr(PALETTE(1))
palette+4	&(palette[2])	Varptr(PALETTE(2))
palette+6	&(palette[3])	Varptr(PALETTE(3))
palette+8	&(palette[4])	Varptr(PALETTE(4))
palette+10	&(palette[5])	Varptr(PALETTE(5))
.....		
.....		
palette+62	&(palette[31])	Varptr(PALETTE(31))

L'indirizzo di ciascun elemento viene calcolato sommando l'indirizzo dell'array con l'indice dell'elemento in questione, moltiplicato per 2 (questo perché una word occupa 2 byte). Ecco un frammento di programma che carica in D1 il contenuto della variabile palette[4]:

```
LEA    palette,A0    ;A0 punta all'inizio dell'array
MOVEQ  #4,D0         ;elemento palette[4]
MULU.W #2,D0         ;D0 = D0 * 2
MOVE.W 0(A0,D0.L),D1 ;MOVE palette+8,D1
```

L'istruzione MULU esegue la moltiplicazione D0*2 e memorizza il risultato in D0. L'istruzione successiva utilizza il modo d'indirizzamento ARI con offset e indice (in questo caso l'offset potrebbe anche omettersi, in quanto è zero): la word contenuta all'indirizzo A0+D0 viene caricata in D1. In maniera molto simile sarà possibile gestire vettori a più dimensioni.

Le strutture e gli stack frame

Mentre i dati di un array hanno tutti le stesse dimensioni, una struttura (chiamata record in Pascal e Modula 2) è un insieme di dati non necessariamente dello stesso tipo. I dati all'interno di una struttura vengono chiamati campi o parametri. In Assembly, analogamente a quanto detto riguardo ai vettori, una struttura può essere rappresentata tramite un buffer nel quale sono memorizzati i dati che costituiscono la struttura stessa.

Vediamo subito una tipica struttura in Assembly:

```
screen_def:
    DC.W 0,0          ;1° e 2° campo
    DC.W 320,256      ;3° e 4° campo
    DC.W 5            ;5° campo
    DC.B 0,1          ;ecc.
    DC.W 0
    DC.W 15
    DC.L 0            ;10° campo
    DC.L title        ;ecc.
    DC.L 0
    DC.L 0            ;13° e ultimo campo
```

Il primi cinque campi di questa struttura sono dati di tipo word; seguono due byte, due word e infine quattro long word. Se abbiamo contato bene, sono 32 byte in tutto. Questa struttura è già inizializzata, nel senso che ciascun campo ha già un valore ben definito. Supponiamo adesso di voler indirizzare un particolare campo di questa struttura, per esempio il decimo campo, che ha la dimensione di una long word. Il problema è il medesimo già incontrato in precedenza, quando volevamo accedere all'elemento palette[4]. Comunque, nel caso delle strutture, dal momento che i campi non sono tutti dello stesso tipo, occorrerà conoscere a priori la distanza (espressa in byte) che intercorre tra l'indirizzo della struttura e l'indirizzo del campo stesso. L'indirizzo del decimo campo è quindi screen_def + 16. In altre parole, il campo dista 16 byte dall'indirizzo iniziale della struttura, come si può facilmente verificare (a tale scopo basta eseguire la differenza tra i due indirizzi). Possiamo sfruttare efficacemente questa informazione utilizzando il modo di indirizzamento ARI con offset (in questo caso l'offset è 16)...

```
LEA    screen_def,A0 ;punta all'inizio della struttura
MOVE.L 16(A0),D0     ;copia il 10° campo in D0
```

La struttura di dati che abbiamo visto è chiamata NewScreen e viene impiegata per definire un nuovo schermo *Intuition* (simile allo schermo del *Workbench*). In questa struttura ogni campo ha un significato ben preciso. Il decimo campo di questa struttura viene chiamato ns_Font. Praticamente, è possibile utilizzare questo nome simbolico come offset. Ovviamente, la definizione di questo simbolo dev'essere stata precedentemente comunicata all'assemblatore tramite un'istruzione di equivalenza:

```
ns_Font EQU 16
```

È quindi possibile migliorare la leggibilità del programma scrivendo:

```
MOVE.L ns_Font(A0),D0
```

Le definizioni simboliche di tutti gli offset delle varie strutture impiegate dal sistema operativo dell'Amiga si trovano nei file include. Grazie alle tabelle di offset l'accesso alle strutture diventa molto più semplice, e inoltre il sorgente diventa assai più

Figura 1: Rappresentazione grafica di un array

Array "palette", 32 elementi di tipo word

Indirizzi di memoria crescenti	Locazione	Contenuto	Nome di variabile
	n	dato n° 1	palette[0]
	n+2	dato n° 2	palette[1]
	n+4	dato n° 3	palette[2]
	n+6	dato n° 4	palette[3]
	n+8	dato n° 5	palette[4]
	...	...	...
	...	...	...
	...	...	...
	n+62	dato n° 32	palette[31]

leggibile. La Figura 2 mostra la tabella di offset associata alla struttura NewScreen.

Le istruzioni LINK e UNLK consentono al programmatore la realizzazione degli stack frame tipici dei linguaggi di alto livello. Per esempio, è noto che in linguaggio C ogni procedura possiede le proprie variabili "locali", cioè variabili che esistono solo per tutto il tempo durante il quale la procedura è attiva. Alla fine della procedura le variabili locali vengono distrutte. Il comando LINK permette di creare uno stack locale. Come primo passo si salva il puntatore allo stack in un registro indirizzi, in modo da poterlo recuperare più tardi. Successivamente, si fa puntare il registro SP a una nuova zona di memoria. L'istruzione UNLK ricostruisce lo stato iniziale, ripristinando il vecchio puntatore allo stack. Per esempio, l'istruzione:

```
LINK    A5,#-200
```

equivale alla sequenza d'istruzioni:

```
MOVE.L  A5,-(SP)    ;salva il registro A5
MOVE.L  SP,A5       ;salva il vecchio SP
ADD.L   #-200,SP    ;carica il nuovo SP
```

Ora la procedura ha il suo stack "locale", che si trova a 200 byte di distanza dal precedente. A causa del funzionamento dello stack, che, come già detto, cresce a ritroso, in genere si utilizza un offset negativo. In altre parole, lo stack locale di una procedura viene a trovarsi all'interno dello stack appartenente all'unità chiamante. Al termine della procedura si dovrà ripristinare il puntatore al vecchio stack. Ciò potrà essere fatto mediante una:

```
UNLK    A5
```

che formalmente corrisponde alle due istruzioni:

```
MOVE.L  A5,SP       ;ripristina il vecchio stack
MOVE.L  (SP)+,A5     ;ripristina il registro A5
```

È interessante notare che dopo l'istruzione LINK il registro A5 contiene una copia del vecchio stack. In questo modo, la procedura potrà accedere facilmente ai dati inseriti nello stack dal programma chiamante. Ed ecco spiegato in che modo i linguaggi di alto livello realizzano il passaggio degli argomenti a una procedura! Per contro, in linguaggio Assembly il passaggio dei parametri a una procedura, nonché l'eventuale restituzione di un valore di ritorno al programma principale, in genere ha luogo mediante l'utilizzo dei registri della CPU e non tramite lo stack, principalmente per motivi di efficienza.

Le librerie dell'Amiga

Tra l'altro, il passaggio dei parametri tramite registri viene

adoperato anche con le funzioni di libreria del ROM Kernel. Le librerie del sistema operativo giocano un ruolo fondamentale nella programmazione dell'Amiga. Per questo motivo, prima di dedicarci alla stesura di programmi in linguaggio in Assembly, è necessario spendere alcune parole in proposito.

Le librerie del ROM Kernel dell'Amiga sono moduli software costituiti da insiemi di subroutine scritte dai progettisti dell'Amiga. Lo scopo di tali librerie era quello di assicurare la compatibilità tra le diverse versioni del sistema operativo e l'interfacciamento con l'hardware. I programmatori possono utilizzare le funzioni di libreria con i propri programmi, e non solo usando l'Assembly, ma anche con molti altri linguaggi. Queste librerie sono del tipo run-shared, nel senso che i codici oggetto delle varie funzioni non vengono "fusi" insieme al codice dei nostri programmi (a differenza delle librerie di tipo linked), ma restano in memoria a disposizione di tutti i programmi che vogliano farne uso. Le principali librerie dell'Amiga sono:

- Exec - gestione multitasking e gestione delle risorse.
- Graphics - grafica e animazione.
- DOS - gestione dei dispositivi di I/O.
- Intuition - gestione dell'interfaccia utente punta e clicca.
- Layers - gestione delle bitmap multistrato.
- Icon - gestione dell'interfaccia a icone (*Workbench*).

Ogni libreria possiede decine di funzioni, ciascuna delle quali svolge un compito particolare. Per esempio, nella libreria DOS esiste una funzione chiamata Rename che consente di cambiare il nome a un file. In questo corso non potremo occuparci in dettaglio delle funzioni di libreria (le principali librerie dell'Amiga contengono insieme oltre 300 funzioni!): ci limiteremo a descrivere le modalità d'uso delle librerie di sistema in linguaggio Assembly. Per chi fosse interessato alle funzioni del ROM Kernel consigliamo la lettura del manuale *Programmare l'Amiga Vol. 1* di Eugene P. Mortimore, edito dalla IHT Gruppo Editoriale. Questo libro, in vendita nelle migliori librerie (distribuito dalla Rizzoli Libri) e in molti negozi di hardware/software, illustra dettagliatamente tutte le principali funzioni di sistema, ed è quindi uno strumento indispensabile per chiunque voglia sfruttare al massimo le potenzialità offerte dal sistema operativo dell'Amiga. Chi avesse difficoltà nel rintracciare questo testo può servirsi dell'apposito modulo di ordinazione presente alla fine della rivista (a pagina 95).

I moduli di libreria si distinguono in moduli residenti e non residenti. Le librerie residenti si trovano in ROM e sono quindi immediatamente disponibili. I moduli non residenti si trovano sul dischetto di sistema nella directory logica LIBS: e devono essere caricati in memoria all'occorrenza. Comunque, il programmatore non deve occuparsi di questa differenza: tutto ciò che deve fare è richiamare la funzione OpenLibrary, specificando come parametri in ingresso un puntatore al nome della libreria (in A1) e la versione di libreria richiesta (in D0). Nel caso di librerie non residenti, la funzione si preoccupa, in maniera del tutto trasparente all'utente, di caricare il modulo di libreria nella RAM.

Com'è organizzata una libreria? In linea di massima, possiamo dire che un modulo di libreria è formato da una tabella di vettori di salto, da un nodo di libreria e da un'area dati (Figura 3). Il nodo di libreria è rappresentato da una struttura Library che viene utilizzata per mantenere la libreria nella lista di librerie del sistema. In questa struttura si trovano altre utili informazioni, come la versione e il numero di revisione della libreria. L'area dati contiene le variabili globali usate dalle funzioni della libreria e, in alcuni casi, il corpo stesso delle funzioni. L'ultimo elemento del modulo di libreria, che poi è quello che ci interessa maggiormente, è la tabella dei vettori di salto. Ogni vettore di

Figura 2: La tabella di offset della struttura NewScreen

Etichetta dell'offset	Valore dell'offset	Dimensione campo	Significato
ns_LeftEdge	EQU 0	W	Coordinata x
ns_TopEdge	EQU 2	W	Coordinata y
ns_Width	EQU 4	W	Larghezza schermo
ns_Height	EQU 6	W	Altezza schermo
ns_Depth	EQU 8	W	Numero di bitplane
ns_DetailPen	EQU 10	B	Colore titolo schermo
ns_BlockPen	EQU 11	B	Colore barra di trascinamento
ns_ViewModes	EQU 12	W	Modo grafico
ns_Type	EQU 14	W	Tipo di schermo
ns_Font	EQU 16	L	Fonte associata
ns_DefaultTitle	EQU 20	L	Titolo schermo
ns_Gadgets	EQU 24	L	Gadget associati
ns_CustomBitMap	EQU 28	L	Bitmap associata

Dimensione totale della struttura (espressa in byte):
ns_SIZEOF EQU 32

salto è associato univocamente a una funzione. La tabella di vettori non è altro che un array, i cui elementi (vettori di salto) sono istruzioni di salto incondizionato (JMP). In sostanza, un vettore di salto è un'istruzione del tipo JMP \$xxxxxxx (ogni vettore occupa 6 byte), dove \$xxxxxxx è l'indirizzo di memoria a partire dal quale inizia la funzione associata. Notiamo che i vettori sono memorizzati in ordine inverso, come se si trovassero nello stack (il primo vettore si trova alla fine della tavola). Per richiamare una funzione di libreria, bisogna anzitutto procurarsi l'indirizzo base della library, che poi coincide con l'indirizzo del nodo di libreria. A questo punto, è sufficiente "saltare" all'indirizzo nel quale è memorizzato il vettore corrispondente alla funzione che si intende chiamare. Per fare ciò, si utilizza un'istruzione JSR \$xx(A6), nella quale riconosciamo l'indirizzamento ARI con offset. Il registro A6 deve contenere l'indirizzo base della library, mentre "xx" è l'offset che ci permette d'individuare il vettore di salto. Da quanto abbiamo detto, la tavola dei vettori di salto precede il nodo di libreria, e quindi questo offset sarà un numero negativo. L'unica libreria che non necessita di essere aperta è l'Exec, il cui indirizzo base è contenuto all'indirizzo \$00000004 (_SysBase).

Ecco il codice Assembly che apre la libreria DOS:

```
lea    dosname,a1    ; puntatore al nome
moveq  #0,d0         ; versione
move.l  $4,a6        ; base di exec
jsr     -552(a6)     ; chiama OpenLibrary
move.l  d0,dosbase   ; salva base di dos
.....
dosname: DC.B        'dos.library',0
        EVEN
dosbase: DC.L        0
```

Questo stesso codice può essere utilizzato per aprire qualunque libreria di sistema: basta cambiare il nome della libreria; quest'ultimo *deve* sempre terminare con un byte di zero (stringhe a terminazione nulla). D'altra parte, questo accade anche in C, dal momento che il compilatore si occupa automaticamente di terminare tutte le stringhe con un carattere NULL (codice ASCII 0).

L'istruzione JSR richiama la funzione OpenLibrary: il vettore corrispondente si trova a una distanza di -552 byte dalla base dell'Exec. Di solito, nei programmi in Assembly si utilizzano dei nomi simbolici per indicare gli offset dei vettori libreria, proprio come per gli offset delle strutture. Per esempio, se all'inizio del programma dichiariamo:

```
_LVOOpenLibrary: EQU -552
```

successivamente potremo usare questo nome simbolico al posto dell'offset numerico:

```
jsr     _LVOOpenLibrary(a6)
```

il che è senza dubbio molto più leggibile (_LVO sta per Library Vector Offset). Allo stesso modo, l'indirizzo \$00000004 potrebbe essere indicato mediante una label:

```
_SysBase: EQU $4
```

La funzione OpenLibrary restituisce in D0 la base della libreria, dunque memorizziamo questo valore nella locazione dosbase. La base ci servirà in futuro ogni qualvolta dovremo richiamare una funzione della libreria DOS...

```
move.l  dosbase,a6    ;base di dos
jsr     _LVOOutput(a6) ;chiamata di una
                        ;funzione dos
```

Alla fine del programma, sarà necessario richiudere la libreria,

per restituire al sistema la memoria da essa occupata. A tale scopo la libreria Exec ci mette a disposizione la routine CloseLibrary (offset -414). L'unico parametro da passare a questa funzione è l'indirizzo base della libreria che intendiamo chiudere, che dev'essere memorizzato in A1. Vediamo dunque il codice Assembly che richiude la libreria DOS:

```
move.l  dosbase,a1    ;base di dos
move.l  _SysBase,a6    ;base di exec
jsr     _LVOCloseLibrary(a6) ;chiude la dos lib
```

Ci sono alcune regole che tutti i programmatori in Assembly dovrebbero seguire per un corretto utilizzo delle funzioni di libreria:

- 1) La base della libreria dev'essere sempre memorizzata in A6.
- 2) Gli argomenti delle funzioni devono essere caricati nei registri appropriati. Le routine che restituiscono un valore lo memorizzano in D0.
- 3) I registri D0, D1, A0, A1 devono essere considerati come alterabili. Ciò significa che una funzione di libreria può modificare a suo piacimento il contenuto di questi registri. Gli altri registri vengono invece preservati automaticamente da tutte le funzioni di libreria.

Ciascuna funzione di libreria ha una propria sintassi. La sintassi della funzione OpenLibrary è la seguente:

```
library = OpenLibrary(libname, version)
D0              A1              D0
```

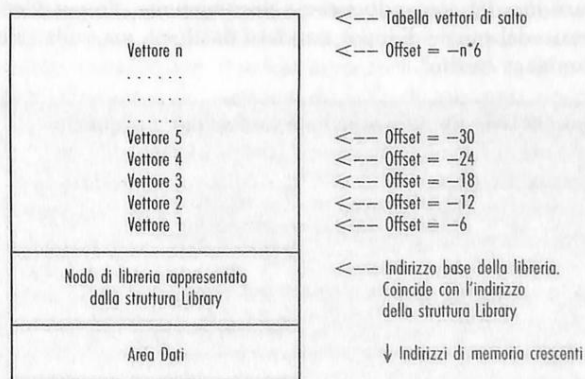
I parametri libname e version devono essere passati alla funzione caricandoli nei registri A1 e D0.

- libname è l'indirizzo della stringa a terminazione nulla che rappresenta il nome della libreria (si raccomanda di utilizzare caratteri in minuscolo).
- version è la versione di libreria che s'intende aprire. A ogni nuova versione del Kickstart, le librerie vengono riscritte e possibilmente migliorate. Ecco i codici di versione più comuni:

```
0      versione indifferente
33     Kickstart 1.2
34     Kickstart 1.3
36     Kickstart 2.00
37     Kickstart 2.04
```

Se la libreria richiesta esiste, e se la versione di libreria è maggiore o uguale alla versione richiesta, la funzione restituisce in D0 l'indirizzo base della libreria. Se la funzione fallisce viene restituito un valore nullo. Vediamo ora la sintassi della funzione CloseLibrary:

Figura 3: Struttura di un modulo di libreria




```
CloseLibrary(library)
A1
```

Questa funzione si aspetta in A1 l'indirizzo base della libreria che s'intende chiudere.

I file d'inclusione giocano un ruolo importantissimo nella programmazione dell'Amiga. Questi file sono moduli-sorgenti che vengono caricati in fase di assemblaggio e uniti al codice sorgente principale (tramite la direttiva INCLUDE). In questi file si trovano le definizioni degli offset delle strutture, degli offset dei vettori di libreria (LVO), e le definizioni di tutte le costanti necessarie per un corretto utilizzo delle funzioni di sistema. Se disponete di questi file include (in genere si trovano sul disco dell'assemblatore), sarà bene stamparli su carta in modo da poterli studiare attentamente. Questi file sono completamente listati nell'*Amiga ROM Kernel Manual - Includes and Autodocs*, edito dalla Addison Wesley.

La ridirezione dell'I/O dall'AmigaDOS

L'AmigaDOS consente di comunicare con i vari dispositivi di I/O. La Figura 4 mostra una lista dei dispositivi più comuni. Si noti che, fatta eccezione per il dispositivo "*", ogni nome di dispositivo termina con i due punti (:). Normalmente, l'input proviene dalla tastiera, mentre l'output viene inviato alla finestra CLI corrente. Per esempio, digitando il comando DIR nella finestra CLI corrente si potrà ottenere un lista delle directory e dei file presenti su disco. L'output viene inviato alla finestra CLI corrente, ma si potrebbe volerlo inviare da qualche altra parte, per esempio alla stampante. Niente di più facile, basta digitare:

```
DIR > PRT:
```

Il simbolo ">" è chiamato comando per la ridirezione dell'output. Esso consente di dirigere l'output verso un altro dispositivo diverso da quello di default. Un altro esempio è dato dalla seguente riga di comando:

```
DIR > DF0:dir.lst
```

Ciò provoca la creazione del file "dir.lst" sul dispositivo DF0. L'output che era destinato alla finestra CLI è stato dirottato verso il dispositivo DF0. Per una verifica provate a digitare:

```
TYPE DF0:dir.lst
```

Per quanto riguarda l'input standard, esso proviene normalmente dalla tastiera, ma anche qui è possibile accettare dati in input da un dispositivo diverso, utilizzando il simbolo di ridirezione dell'input "<". Com'è noto, il comando AmigaDOS:

```
TYPE ?
```

consente d'introdurre i parametri del comando durante l'esecuzione del comando stesso. Normalmente, l'input viene prelevato dal canale di input standard (tastiera), ma nulla vieta di "cambiare canale"...

Figura 4: Lista dei dispositivi più comuni dell'AmigaDOS

Nome	Descrizione
DF0:	drive interno o prima unità disk drive
DF1:	drive esterno o seconda unità disk drive
RAM:	ram disk
PRT:	stampante
PAR:	dispositivo connesso alla porta parallela
SER:	dispositivo connesso alla porta seriale (ad es. modem)
CON:	finestra CLI con input filtrato (permette l'editing)
RAW:	finestra CLI con input grezzo (non permette l'editing)
*	finestra CLI corrente

```
TYPE < CON:0/0/640/50/Input_Window ?
```

Come vedete, l'input viene prelevato da un'altra "console". Da ciò risulta che ogni finestra CLI possiede un proprio canale di input e di output.

L'I/O da linguaggio Assembly

Abbiamo visto come normalmente l'AmigaDOS riceva e invii le informazioni attraverso i canali di input e output standard. Ogni qualvolta si usa la ridirezione (o quando si usa il comando NEWCLI), l'AmigaDOS apre un nuovo canale per il flusso dei dati, e si preoccupa di richiuderlo automaticamente alla fine delle operazioni. Per fare ciò, l'AmigaDOS crea in memoria una struttura FileHandle (definita nel file include dosextens.i), che, in sostanza, rappresenta il "canale di comunicazione" tra il dispositivo e il flusso delle informazioni (file). Di conseguenza, un programma che voglia eseguire una qualunque operazione di I/O dovrà necessariamente conoscere l'indirizzo della struttura FileHandle associata al canale di I/O. Questa informazione può essere ottenuta utilizzando la funzione Open della libreria DOS, la cui sintassi è la seguente:

```
file = Open(name, accessmode)
D0          D1    D2
```

La funzione Open richiede due parametri in ingresso:

1) L'indirizzo di una stringa a terminazione nulla in D1. Questa stringa può essere: un nome di file (eventualmente preceduto da un nome di device, per esempio "DF0:"); un semplice nome di dispositivo (come "PRT:" o "NIL:"); una descrizione di console del tipo "CON:" o "RAW:", seguita dai parametri di finestra; il device "*", che rappresenta la finestra corrente.

2) Il modo di apertura del file, MODE_OLDFILE o MODE_NEWFILE, in D2 (questi simboli sono definiti nel file include dos.i). Il primo viene usato per aprire un file già esistente per la lettura o la scrittura, mentre il secondo crea un nuovo file per la scrittura, assegnandogli il nome indicato dal primo parametro.

La funzione restituisce in D0 l'indirizzo della struttura FileHandle usata per la gestione dell'I/O. Se per qualche motivo non è stato possibile aprire il file, la funzione restituisce zero. Nel caso di input/output da e verso i canali di I/O standard non si ricorrerà mai alla funzione Open (infatti questi canali sono già aperti). Le funzioni Input e Output della libreria DOS possono essere utilizzate per ottenere gli indirizzi delle strutture FileHandle associate rispettivamente al canale di input standard (_stdin) e al canale di output standard (_stdout). Entrambe le funzioni non richiedono nessun parametro e restituiscono l'indirizzo della struttura FileHandle in D0.

Una volta che il canale di I/O è aperto, si possono eseguire due operazioni: è possibile inviare dati al dispositivo (output) oppure riceverli da esso (input). Per inviare un flusso di dati verso il dispositivo, si utilizza la funzione Write:

```
returnedLength = Write(file, buffer, length)
D0              D1    D2    D3
```

Il parametro file è l'indirizzo della struttura FileHandle associata al dispositivo verso cui s'intende inviare il flusso di dati. Il parametro buffer è l'indirizzo dell'area di memoria nella quale sono contenuti i dati da inviare al dispositivo. Il parametro length indica le dimensioni del buffer di memoria, espressa in byte.

Se l'operazione di scrittura ha successo, viene restituito il numero di byte trasmessi (che poi coincide con il parametro

length). Diversamente, viene restituito un valore di -1. Viceversa, è possibile ottenere informazioni da un dispositivo mediante una chiamata alla funzione Read.

```
actualLength = Read(file, buffer, length)
D0          D1    D2    D3
```

Il parametro file è l'indirizzo della struttura FileHandle associata al dispositivo dal quale s'intende leggere il flusso di dati. Il parametro buffer è l'indirizzo dell'area di memoria destinata ad accogliere i dati in arrivo dal dispositivo. Il parametro length indica il numero di byte (per esempio caratteri ASCII) da leggere. Ovviamente, questa lunghezza non deve superare le dimensioni del buffer di memoria.

Se l'operazione di lettura ha successo, viene restituito il numero di byte letti. Se la funzione restituisce un valore nullo, ciò sta a indicare che è stata raggiunta la fine del file (condizione

Il primo vero programma in Assembly

Il programma che sto per presentarvi non fa nulla di particolarmente eccitante, tuttavia può aiutarci a comprendere meglio l'utilizzo delle funzioni di libreria del sistema operativo dell'Amiga e la filosofia dell'I/O su cui è basato l'AmigaDOS. Per il momento, tutti i programmi che scriveremo dovranno essere eseguiti da CLI. Non sarà infatti possibile eseguire i nostri programmi da Workbench (ciò infatti richiederebbe il link con il codice di start-up...). Il programma chiede all'utente di digitare un testo e quindi lo memorizza nel RAM Disk. Osserviamo attentamente la sintassi delle istruzioni e il formato del codice sorgente (Figura 5). Assicuratevi che il vostro assembler sia configurato per distinguere le maiuscole dalle minuscole. Nei listati ho usato dei nomi ben precisi per gli offset e per le costanti. In teoria, nulla vieterebbe di scrivere openlib al posto

Figura 5

```
incdir    "include:"
include   exec/exec_lib.i
include   libraries/dos_lib.i
MODE_NEWFILE EQU 1006 ;oppure INCLUDE
;libraries/dos.i

* apre la libreria DOS
lea       dosname,a1 ;nome
moveq     #0,d0      ;versione
move.l    $4.w,a6     ;base di Exec
jsr       _LVOOpenLibrary(a6)
tst.l     d0          ;tutto OK ?
beq       cleanup    ;no
move.l    d0,_DOSBase ;si

* ottiene gli handler di input e output standard
move.l    d0,a6       ;base di DOS
jsr       _LVOOutput(a6)
move.l    d0,_stdout  ;salva _stdout
jsr       _LVOInput(a6)
move.l    d0,_stdin   ;salva _stdin

* invia testo su _stdout
move.l    _stdout,d1  ;file handle
move.l    #text,d2    ;buffer
move.l    #(textlen-text),d3 ;lunghezza
jsr       _LVOWrite(a6)

* legge input da _stdin
move.l    _stdin,d1   ;file handle
move.l    #buffer,d2  ;buffer
move.l    #255,d3     ;lunghezza
jsr       _LVORRead(a6)
tst.l     d0          ;tutto OK ?
ble.s     cleanup    ;no
move.l    d0,len      ;si

* apre file su RAM Disk
move.l    #filename,d1 ;nome file
move.l    #MODE_NEWFILE,d2 ;write mode
jsr       _LVOOpen(a6)
tst.l     d0          ;tutto OK ?

beq.s     cleanup    ;no
move.l    d0,file     ;si
* scrive file su RAM Disk
move.l    d0,d1       ;file handle
move.l    #buffer,d2  ;buffer
move.l    len,d3       ;lunghezza
jsr       _LVOWrite(a6)

* chiude file
cleanup:
move.l    file,d1
cmp.l     #0,d1       ;file aperto ?
beq.s     cdl         ;no
jsr       _LVOClose(a6) ;si

* chiude DOS library
cdl:
move.l    _DOSBase,a1
cmp.l     #0,a1       ;DOS library aperta ?
beq.s     exit        ;no
move.l    $4.w,a6     ;base di Exec in a6 e
jsr       _LVOCloseLibrary(a6) ;chiusura della
;DOS library

* torna al CLI
exit:     rts          ;fine

*****
* Zona dati *
*****
_stdin:   dc.l 0
_stdout:  dc.l 0
_DOSBase: dc.l 0
file:     dc.l 0
len:      dc.l 0
dosname:  dc.b 'dos.library',0
even
filename: dc.b 'RAM:testo.asc',0
even
text:     dc.b 'Digitare il testo (max. 255 chars): '
textlen:  even
buffer:   dc.b 255
even
```

di EOF). In caso di errore viene restituito un valore di -1.

Quando abbiamo terminato le operazioni di I/O è molto importante non dimenticare di richiudere il canale di comunicazione. Per fare ciò, utilizziamo la funzione Close:

```
success = Close(file)
D0          D1
```

Il parametro file è un puntatore alla struttura FileHandle ottenuto dalla precedente chiamata alla funzione Open. Esaminando il valore di ritorno della funzione, è possibile stabilire se, per un qualsiasi motivo, il programma non sia stato in grado di chiudere il file con successo (success = 0). Si noti che la funzione Close restituisce un valore soltanto a partire dalla versione 36 della libreria DOS (Kickstart 2.00). Nelle versioni precedenti la funzione non restituiva alcun valore. I canali di I/O standard non dovranno mai essere chiusi.

di _LVOOpenLibrary. Tuttavia, in questo caso non sarebbe possibile utilizzare i file include. Sarà bene quindi utilizzare nei propri programmi gli stessi nomi così come compaiono nei file include. Inizialmente, il programma apre la libreria DOS nella quale si trovano le funzioni di I/O di cui esso necessita. Successivamente, il programma ottiene gli indirizzi dei FileHandle "_stdin" e "_stdout" e apre un canale di output verso il RAM disk. terminate le operazioni il canale di output viene richiuso tramite una Close. Prima di tornare al CLI, il programma chiude la libreria DOS, compito che viene affidato alla funzione CloseLibrary.

Come esercizio, provate a modificare il programma in modo che il testo venga inviato alla stampante. Successivamente, fate in modo che il testo in ingresso sia prelevato da un file ASCII memorizzato su disco.

LE ESTENSIONI AMOS AL MICROSCOPIO

Tutto quello che dovete sapere per realizzare un'estensione per AMOS

a cura di Avelino De Sabbata

Questa è una rubrica fissa dedicata ad AMOS. Potete inviare i vostri programmi per l'inserimento nella raccolta di pubblico dominio IAPP, porre domande di carattere tecnico, inviare stratagemmi di programmazione, critiche... Potete insomma partecipare attivamente alla stesura di queste pagine scrivendo all'indirizzo riportato in fondo all'articolo.

Non sono pochi i programmatori che essendo in grado di programmare in assembly avrebbero voluto personalizzare e potenziare l'ambiente AMOS. Purtroppo, le tecniche per aggregare le singole routine in una libreria e interfacciare il codice macchina ad AMOS erano descritte da Lionet in modo insufficiente e lacunoso. Finalmente, grazie agli sforzi e alle sperimentazioni dell'amico Stefano Peruzzi chiunque potrà implementare nuovi comandi in AMOS: come vi avevamo preannunciato, infatti, questo *Spazio AMOS* è dedicato proprio a questo argomento estremamente interessante.

Le estensioni di AMOS

Ogni programma, indipendentemente dal linguaggio con cui è scritto, richiede sempre l'ideazione di una serie di algoritmi e procedure, la definizione delle strutture dati, la traduzione di tutto ciò nel linguaggio scelto e infine una lunga fase di debug; se poi si calcola il tempo necessario per la realizzazione del programma, dalla sua prima ideazione alla fase ultima di debug, si scopre che è come se si scrivessero circa 10 righe al giorno! Questa considerazione è significativa proprio perché il dato è indipendente dal linguaggio utilizzato e se con 10 righe in assembly si può aprire e chiudere una run time library, in AMOS si può anche fare un programma completo!

In effetti, molti programmatori, specialmente quelli che usano l'assembly, hanno sempre sotto mano una lunga serie di routine sviluppate per risolvere un particolare problema ed è in questo modo che possono scrivere più rapidamente un programma, preoccupandosi solo di modificare, aggiungere o interfacciare queste routine. Questa è anche la filosofia di AMOS, un ambiente di programmazione che permette a chiunque di utilizzare una vasta libreria di routine scritte proprio da Francois Lionet, i famosi 500 e più comandi del linguaggio. L'interprete, e ancor più il compilatore, non fanno altro che interfacciare tra loro tutte queste routine automaticamente, riducendo di fatto il tempo di stesura e di debug. Ma AMOS fa molto di più, permette infatti a chiunque di estendere l'intero set di comandi aggiungendo routine nuove e personalizzate: un'estensione è proprio questo.

La possibilità di scrivere estensioni era già presente con AMOS

1.1 e, non a caso, già allora il pacchetto ne comprendeva alcune. Con la versione 1.34 Lionet ha però modificato il formato dei dati e ha stabilito nuove regole per la stesura del codice di libreria. Con questo articolo tenterò di spiegare proprio come si sviluppa una di queste estensioni sulla base dell'esperienza fatta durante la stesura prima della Synapsis.Lib e ultimamente delle estensioni ARExx e Math.

Comincio col precisare che è assolutamente necessario utilizzare l'assembly, quindi niente C, Pascal o altro; a questo proposito colgo l'occasione per invitare tutti a fare delle prove in assembly. Anche in questo caso AMOS è un ottimo insegnante, ecco come fare:

- 1) Pensate a una routine utile e sviluppatela poi in assembly (ovviamente dovete imparare a farlo con un buon libro sul 68000).
- 2) Compilatela e procedete al link come per il codice eseguibile. A questo punto normalmente non avreste ancora niente in mano, questo perché al vostro programma mancherebbe l'interfaccia I/O: le procedure per introdurre i parametri in ingresso e quelle per comunicare il risultato.
- 3) Caricate il vostro codice eseguibile con Pload da AMOS e mettetelo in una bank, per esempio la 15.
- 4) Passate alla routine tutti i parametri necessari con i comandi Areg, Dreg e infine eseguirla con il comando Call.

In breve, potrete scrivere il vostro codice e vederlo funzionare anche se è la prima volta che usate l'assembly, questo senza preoccuparvi dell'interfaccia con il sistema operativo, operazione che richiede una buona conoscenza del sistema. Il metodo appena descritto può servirvi anche durante la stesura dell'estensione per provare le routine prima d'includerle nella stessa.

Un altro punto da chiarire è quello sull'assemblatore: ne esistono diversi sia di commerciali, sia di pubblico dominio, e anche uno scritto in AMOS e incluso nel pacchetto del compilatore stesso. Personalmente, sconsiglio il *Seka* e consiglio vivamente il *Devpac 2*, o meglio il *Devpac 3*, della HiSoft che ritengo sia il migliore sul mercato, d'ora in poi mi riferirò sempre a questo.

Un'estensione AMOS non è un programma eseguibile, ma una struttura dati, cioè una libreria, che viene poi utilizzata dall'interprete o dal compilatore; non provate mai quindi a dare il run a un'estensione, pena un guru per istruzione non legale (tipo 4). A differenza del sistema operativo, AMOS sa bene cosa fare con i dati presenti in questa libreria non standard, ma solo se questi dati rispettano un preciso formato.

Fondamentalmente, un'estensione si compone di quattro parti. Una prima contenente una tabella di offset a tutte le routine dell'estensione, cioè una serie d'indici che indicano ad AMOS gli spiazziamenti dall'indirizzo in memoria a cui sarà caricata l'estensione a quello dell'inizio effettivo delle routine. Se per esempio un offset vale \$100, ciò significa che se l'estensione verrà caricata alla locazione \$50000 la routine in questione comincerà dalla locazione \$50100 esadecimale, se invece l'estensione verrà caricata alla locazione \$80000 la routine comincerà alla \$80100. Una seconda parte con la lista di tutti i token, cioè con la lista di tutti i comandi che l'editor e il compilatore dovranno riconoscere assieme al tipo dei loro parametri. Una terza parte contenente gli eventuali dati utili e tutte le routine, incluse quelle che AMOS chiama ogni volta che carica l'estensione (cioè quando si carica AMOS stesso) o che la cancella (quando si esce da AMOS), quelle dei nuovi comandi e le ultime due che riguardano sempre la gestione degli errori. Infine, la quarta e ultima parte, la più semplice, con l'indicazione della scritta di copyright che deve apparire all'inizio.

Prima di cominciare la descrizione dettagliata di queste quattro parti, è bene premettere che il codice di tutta l'estensione dev'essere completamente rilocabile. Non devono mai esserci indirizzi assoluti e questo, a dire il vero, è un dogma della programmazione sull'Amiga assieme a quelli del codice non automodificante e del rispetto del multitasking. In pratica, il programma deve poter funzionare in qualsiasi posizione di memoria venga caricato; se per esempio dovete caricare in D0 un dato immagazzinato a un indirizzo indicato da una label "Pippo", dovete scrivere "Lea Pippo(pc),A0" per caricare prima quell'indirizzo come offset al program counter in A0, e poi "move.l(a0),d0" per caricare D0 con il dato cercato. Se avete la fortuna di usare il *Devpac 3* potrete impostare l'opzione "Ensure PC relative code" e sarete automaticamente avvisati di ogni errore.

Tecnicamente, l'estensione comincia come ogni programma Amiga (anche se poi non è eseguibile direttamente) con un Header Hunk seguito da un unico Code Hunk, lo dico per far notare che l'estensione dev'essere costituita da un unico hunk (più quello di Header); ovviamente, tutto questo lo fa automaticamente l'assemblatore, l'importante è che controlliate come si comporta il vostro. Nel caso del *Devpac*, se non specificate il contrario, il codice viene assemblato e linkato in un hunk soltanto; se avete dei dubbi, dopo aver assemblato l'estensione provate a controllare con un "type opt h" da *CLI*: dopo la prima long word con il codice \$3f3 e quella successiva, la terza long word dovrebbe riportare il valore \$00000001 (il 12 byte \$01) per indicare proprio la presenza di un solo hunk. All'inizio del listato potrete cominciare col definire una costante molto usata *ExtNb* (il numero dell'estensione), precisamente *ExtNb=NumeroExt-1*, se avete deciso di caricare l'estensione col numero 20 dovete definire *ExtNb=19*. *ExtNb* verrà poi utilizzato spesso e sarà così impossibile cambiare il numero dell'estensione una volta che sia stata assemblata, a meno di non tener traccia di dove si è usato quel numero e modificare poi direttamente il codice oppure più semplicemente di compilare più copie dell'estensione cambiando solo *ExtNb*. A questo punto si possono includere i file forniti con AMOS contenenti molte informazioni utili se desiderate utilizzare le funzioni della libreria principale di AMOS. Il codice vero e proprio comincia con 4 long word: la prima indica l'inizio della lista generale degli offset, la seconda quello della tabella dei token, la terza quello della prima routine, che è anche quella eseguita quando si carica

AMOS, e la quarta quello della stringa che contiene il copyright iniziale. Per motivi di rilocazione, questi punti sono indicati ognuno come differenza tra la label successiva e quella cercata, se allora le quattro sezioni dell'estensione sono indicate in ordine dalle label (etichette) *C_Off*, *C_Tk*, *C_Lib*, *C_Title* e da un'ultima *C_End* che corrisponde all'ultima word effettiva dell'estensione e che deve sempre essere posta a zero, le prime quattro long word conterranno le quattro differenze *C_Tk-C_Off*, *C_Lib-C_Tk*, *C_Title-C_Lib* e *C_End-C_Title*. AMOS per trovare la prima routine dell'estensione sommerà le prime tre long word ottenendo uno spiazziamento rispetto all'indirizzo a cui ha già caricato l'estensione.

Dopo questi quattro offset, segue una word che può essere posta a -1 o 0 a seconda che si desideri che la prima funzione della libreria sia sempre copiata o meno in un programma AMOS dal compilatore. Notate che questo vale solo per la prima routine poiché tutte le altre saranno copiate dal compilatore nell'eseguibile solo se saranno usate almeno una volta nel listato AMOS. Aggiungo che l'interprete funziona diversamente in quanto carica in blocco tutta l'estensione, il compilatore carica invece solo le routine usate dal programma ed è per questo che se internamente all'estensione vorrete chiamare una routine scritta precedentemente, o facente parte della libreria principale di AMOS, dovete usare un modo tutto speciale per farlo, attraverso delle macro definite da Lionet che inseriscono dei codici comprensibili al compilatore. Ricordate quindi che, una volta che un programma AMOS è compilato, non potete fare nessuna congettura sulla posizione di una qualsiasi routine rispetto a un'altra.

Parte I

Cominciamo allora con la tabella dei salti alle routine, che inizia con l'etichetta *C_Off* ed è seguita da una lunga serie di definizioni di campi costanti del formato word, una lunga serie di DC.w che contengono le semidifferenze tra gli indirizzi delle etichette delle routine. Mi spiego meglio: ogni routine che inserite nell'estensione dovrà necessariamente essere compresa tra due etichette e avere una lunghezza in byte pari, questo significa che alla prima istruzione della routine dovete associare una label e questo per ogni routine; così facendo, si avrà che ogni routine sarà delimitata dalla label a essa associata e da quella associata alla routine successiva nel listato, meno l'ultima di tutte per la quale è necessario perciò aggiungere in coda un'ultima etichetta. Se allora le vostre routine nell'ordine sono RO, R1, R2, R3... corrispondenti alle label LO, L1, L2, L3... RO sarà delimitata da LO e L1... Il secondo punto da tenere in considerazione è che la lunghezza di ogni routine dev'essere pari in numero di byte e quindi, qualora utilizzate dei DC.b, ricordatevi sempre di aggiungere un EVEN o un CNOP in coda.

La lista in *C_Off* sarà del tipo DC.w (L1-LO)/2, (L2-L1)/2, (L3-L2)/2... e AMOS ogni volta che avrà bisogno, per esempio, della R3 non farà altro che prendere i primi tre valori, raddoppiarli e sommarli prima tra loro e poi all'indirizzo di *C_Lib*, che corrisponde propriamente a LO e che aveva già ricavato dall'analisi delle prime quattro long word.

Come ho detto, la lista dev'essere completa di tutte le label e andrà aggiornata a mano a mano che aggiungerete funzioni e comandi all'estensione. Tenete presente che la prima routine, la LO, sarà quella eseguita quando parte AMOS e dovrà perciò contenere il vostro codice d'inizializzazione; non dimenticate

poi che l'ultima e la penultima routine devono essere quelle scritte da Lionet stesso per la gestione degli errori (le vedremo poi). In questa lista potrete annotare gli offset di ben 2000 routine, ognuna lunga fino a 128K... non male!

Parte II

L'estensione continua con la token table, cioè con una tavola che indica ad AMOS e al compilatore il nome che avete scelto per ogni vostro comando, il numero e il tipo dei parametri che accetta e nel caso si tratti di una funzione (quest'ultima restituisce un dato), il tipo di dato che può restituire. La tavola deve iniziare con "DC.w 1,0" seguito da un "DC.b \$80, -1". Notate che la prima definisce costanti di tipo W e la seconda di tipo B, nella tabella si alterneranno quindi DC.w e DC.b in quest'ordine, una coppia per ogni comando.

La prima sequenza è costituita da due word, una vale -1 l'altra è il numero della routine da uno in poi; in genere, è comunque più comodo usare un simbolo mnemonico, per esempio per indicare che a un token corrisponde il codice della quinta routine. Potremmo scrivere 5 oppure MioComando, ricordando però di definire "MioComando equ 5" in modo che l'assemblatore quando trova MioComando lo sostituisca proprio con un 5. Per pura convenzione, Lionet ha stabilito che i simboli mnemonici utilizzati in questo punto siano preceduti da una L_ (nell'esempio sarebbe L_MioComando) e che poi si definisca il relativo "L_xxx equ n" prima della routine stessa in modo da utilizzarlo anche come utile traccia mnemonica. Fate anche molta attenzione che il nome mnemonico che avete scelto non coincida con uno di quelli utilizzati da AMOS (si trovano nei file include), in tal caso l'assemblatore vi avviserà di una definizione ripetuta e voi dovreste modificare il vostro simbolo.

Se il token che state descrivendo è un'istruzione, cioè accetta o meno parametri ma non deve comunque restituire alcun valore, allora dovreste prima definire il numero di routine seguito dal -1. Se invece si tratta di una funzione, cioè il comando deve restituire un valore, allora dovreste far precedere -1 al numero della routine. Riassumendo, il primo campo è quello relativo all'istruzione e il secondo alla funzione, un valore di -1 indica che non dev'essere chiamata nessuna routine, altrimenti questo valore è il numero della routine stessa. La seconda sequenza sarà invece assemblata byte per byte (DC.b) e dev'essere formata dal nome completo dell'istruzione/funzione, carattere per carattere (tutti nel rispettivo codice ASCII), con l'ultimo carattere aumentato della costante \$80, deve quindi seguire la lista dei parametri, indicati dai valori ASCII, delle cifre 0,1,2 e delle lettere I e V; questa definizione termina con un byte posto a -1 o a -2. Per quanto riguarda il nome dell'istruzione, è sufficiente scrivere: DC.b "comand", "o" + \$80, il che instruirà AMOS a considerare "comand" come un nuovo token, sempre che in qualche estensione con numero più basso o in AMOS stesso non esista un'istruzione con lo stesso nome che in questo caso avrà precedenza (attenzione ai nomi quindi).

Ora devono seguire i parametri: il primo è sempre riferito al comando e dev'essere I, se si tratta di un'istruzione che non restituisce alcun valore, oppure va sostituito con 0, 1, 2, se si tratta di una funzione che restituisce rispettivamente un intero, un valore float o una stringa caratteri. Dalla seconda posizione in poi si continua con la lista dei parametri veri e propri: in AMOS i parametri possono essere separati da una virgola o da un 'tò, così dovreste fare anche voi inserendo la vostra lista e utilizzando una virgola o una t (minuscola) per il 'tò. I parametri

sono rappresentati dal loro tipo e dalla loro posizione relativa, tocca a voi attribuire poi a ognuno di essi un preciso significato. Per indicare un valore intero, float o una stringa si devono usare appunto le cifre 0, 1, 2 (i valori ASCII di queste cifre); c'è anche il caso di V, seguito da una delle tre cifre, che dovrebbe servire per variabili riservate, ma a questo proposito Lionet non è chiaro e, da prove effettuate, non sembra che funzionino con il compilatore. Ricordate che se due istruzioni hanno lo stesso nome e differiscono però per tipo e posizione dei parametri, AMOS le considererà due istruzioni diverse a tutti gli effetti, infatti tutti i comandi che accettano configurazioni diverse di parametri sono in realtà un insieme di più comandi e così dovreste fare anche voi; se allora dovete specificare un comando che può trattare tre diverse configurazioni di parametri, scrivete le tre definizioni terminando le prime due con un -2 e l'ultima con un -1. La funzione del byte di terminazione è proprio questa: un -1 indica che la definizione termina, mentre un -2 indica che segue un'altra definizione in cui non cambia il nome dell'istruzione, ma soltanto la lista dei parametri. Esiste anche un metodo per risparmiarsi la reintroduzione completa del nome dell'istruzione: è sufficiente che la prima volta si faccia precedere al nome del comando vero e proprio un punto esclamativo ("!comand", "o" + \$80), al momento di reintrodurre il nome basterà scrivere solo \$80 e AMOS capirà. Per terminare la lunga lista delle nuove istruzioni è sufficiente una parola posta a zero (DC.w 0).

Parte III

Arrivati a questo punto si deve procedere all'introduzione delle routine una a una. La prima routine, contrassegnata dalla label LO equivalente all'indirizzo stesso di questa sezione (cioè C_Lib), viene sempre eseguita quando si carica AMOS e deve contenere il vostro codice d'inizializzazione costituito da una prima routine generale e alcune altre più specifiche. La prima dovrebbe essere formulata in questo modo:

```
LO  movem.l  a3-a6, -(sp)
    lea      Dati(pc), a3  Dati, label al blocco dei dati
    move.l   a3, ExtAdr+ExtNb*16(a5)
    lea      Def(pc), a3   Def, label a una routine
    move.l   a3, ExtAdr+ExtNb*16+4(a5)
    lea      Exit(pc), a3  Exit, label a un'altra routine
    move.l   a3, ExtAdr+ExtNb*16+8(a5)
    lea      Bank(pc), a3  Bank, label a una terza routine
    move.l   a3, ExtAdr+ExtNb*16+12(a5)
    *** qui dell' eventuale codice aggiuntivo ***
    movem.l  (sp)+, a3-a6
    moveq    #ExtNb, d0
    rts
```

Seguono le routine:

```
Def  *** codice ***
    rts
Exi  *** codice ***
    rts
Bank *** codice ***
    rts
```

La prima routine, Def, sarà invocata automaticamente ogni volta che caricherete il linguaggio o impartirete il comando Default: qui, per esempio, conviene mettere il codice di

apertura di eventuali librerie o device. La seconda, invece, sarà chiamata alla fine di un programma compilato o quando uscite definitivamente dall'interprete, se avete aperto qualcosa in Def, qui dovete chiuderlo e rilasciarne così la memoria allocata. La terza è invocata ogni volta che si effettua un'operazione su una bank AMOS e serve solo nel caso che la vostra estensione definisca un nuovo tipo di bank (così fa per esempio l'estensione musicale). Dati è la label posta all'inizio del banco dati principale della vostra estensione (sempre che ne serva uno), definendolo a questo modo vi sarà poi facile richiamare un dato in modo rilocabile con due macro che userete spesso e che sono definite nei file include di Lionet: Dlea e Dload.

A proposito di macro, negli include ce ne sono molte altre indispensabili. Per chiamare una routine dell'estensione stessa o una associata a un comando AMOS interno, non potete assolutamente fare un salto diretto, perché la distanza tra le routine varierà ogni volta che il compilatore le copierà in un programma compilato. Proprio per risolvere questo problema, Lionet ha definito una serie di codici comprensibili al compilatore che lo istruiscono su ogni tipo di salto condizionato. In poche parole, a voi basterà utilizzare la macro Rbsr invece di un semplice bsr, seguita da un offset indicante il numero della routine, se cioè da un comando dovete chiamare la routine, relativa a un altro comando, chiamata L_Pippo (già definita come L_Pippo equ 10...), allora vi sarà sufficiente usare una "Rbsr L_Pippo", l'assemblatore tradurrà la macro in \$FE31 \$000A mentre il compilatore capirà che deve inserire in quel punto un salto all'indirizzo effettivo della routine L_Pippo una volta che questa sia stata inserita nel programma compilato. Di queste macro ce ne sono molte, una per ogni possibile salto condizionato (Rbeq, Rbne, Rbcc...), non dimenticate però che gli offset si riferiscono tutti alle routine della vostra stessa estensione; solo in due casi questo non è vero: Rjmp ed Rjsr si riferiscono sempre alla libreria principale di AMOS i cui offset sono ben documentati sempre nei file include.

Quello che ho appena detto può far pensare che in realtà si possa facilmente tradurre un intero programma AMOS in un unico comando che usi tutti gli altri; questo è in teoria possibile, ma non di facile realizzazione, perché gli include riportano solo gli offset e non funzione e parametri dei comandi anche se poi molti sono quelli di AMOS e già si conoscono. Inoltre, chiamando queste funzioni riscritte proprio per il compilatore, non si può essere sempre sicuri che le stesse funzionino anche con l'interprete. Esiste anche un'altra macro speciale, si tratta di Rdata e va sempre messa prima di un grosso blocco di dati: questa macro impedisce al compilatore di confondere un normale dato con uno dei codici speciali di salto, blocca cioè il parser del compilatore fino alla successiva label Lx ed evita così spettacolari crash apparentemente inspiegabili.

Veniamo ora alla stesura di una routine, si tratta di sapere come ricevere e come restituire i parametri. Quando si chiama un comando AMOS seguito dai parametri, questi vengono automaticamente presi e messi in pila uno dopo l'altro a partire dall'indirizzo puntato da A3. Perciò, usando A3 come uno stack pointer AMOS decrementa il suo valore e pone al nuovo indirizzo a cui A3 punta, il primo parametro, e così via fino all'ultimo. Voi dovete perciò prelevare i parametri da questa pila in ordine inverso partendo dall'ultimo che si trova all'indirizzo corrente puntato da A3, incrementando A3 e ripetendo le stesse operazioni fino al primo parametro: attenti a non prelevarne di più o di meno, perché questo manderà sicuramente in crash AMOS. In pratica, dopo l'esecuzione della

vostra istruzione, A3 dovrà puntare esattamente allo stesso indirizzo a cui puntava prima che l'istruzione stessa fosse chiamata. Gli interi e i numeri in virgola mobile (standard FFP) occupano una long word (4 byte) e ci stanno comodamente in un registro interno del 68000, per prelevare parametri di questo tipo usate in ogni caso una "move.l (a3)+,d0"; se invece il parametro è una stringa AMOS vi restituisce un puntatore a essa in (A3) e una word con la lunghezza all'indirizzo puntato da AO. Per prelevare quindi una stringa, recuperate prima il puntatore con una semplice "move.l (A3)+,AO" e poi la sua lunghezza con "move.w (AO)+,DO". In ogni vostra routine potete usare qualsiasi registro macchina, ma non dimenticate di salvare i contenuti di A4, A5, A6, magari con una "movem.l A4-A6,(sp)-" iniziale e recuperarli poi in coda con una "movem.l (sp)+,A4-A6".

Vediamo ora come una funzione restituisce i parametri ad AMOS: caricate in D3 il valore del parametro (se è un intero o un float) o il puntatore a una stringa e, subito dopo, il tipo in D2 (0, 1, 2 come già detto, ma con il valore stesso e non il codice ASCII). Attenzione che, se si tratta di una stringa, questa dovrà iniziare con una word indicante l'effettiva lunghezza seguita dai vari caratteri che la compongono. Le ultime due routine devono sempre essere quelle di controllo degli errori, la penultima va utilizzata quando si compila con l'opzione -E1 o si usa l'interprete, l'ultima quando si compila con l'opzione -EO. Queste le trovate già pronte negli esempi forniti con AMOS: se dovete creare una lista di errori specifici per la vostra estensione, semplicemente definite la lista dei possibili errori, riga su riga, nell'apposito campo dati di L_Custom (appunto la penultima routine), per chiamare poi una di queste stringhe e renderla visualizzabile in caso di errore, basterà assegnare a d0 il numero di errore partendo da zero (moveq #2,DO per il terzo messaggio) e chiamare poi con "Rbra L_Custom" la routine stessa. Aggiungo che, se intendete invece chiamare uno degli errori della lista interna di AMOS, dovete sempre assegnare a D0 il numero dell'errore (quello AMOS questa volta), ma utilizzare poi una "Rjmp L_Error" (L_Error è una delle routine della libreria principale e infatti uso Rjmp e non Rbra!).

Parte IV

Non rimane che introdurre la scritta di copyright, quella corrispondente alla label C_Title:

```
DC.b 31,"La mia prima estensione V1.0",0
even
```

Attenzione a non dimenticare lo "0" come anche in tutte le stringhe che descrivono i vostri errori.

Siamo arrivati alla fine, basta un "DC.w 0" e non rimane che assemblare. L'estensione compilata dovrà sempre chiamarsi xxx.Lib e trovarsi nel cassetto AMOS_System (almeno per praticità), poi, per renderla visibile all'interprete, procedete a caricare il file di configurazione (si usa Config1_3.AMOS) e modificate la lista delle estensioni da caricare inserendo la vostra (per esempio ":Amos_System/xxx.Lib"), attenzione a caricarla col numero giusto. Un ultimo consiglio: osservate bene tutti gli esempi forniti da Lionet, attenti ai guru, ricordate sempre la frase "Only Amiga Makes It Possible" e... buona fortuna. ■

ITALY AMOS USERS CLUB

c/o Avelino De Sabbata
Via G. Carducci, 3 - 33050 Terenzano (UD) - (Tel. 0432/560426)

CORSO DI PROGRAMMAZIONE IN MODULA 2

**Ultima puntata della serie: l'organizzazione
e la gestione delle strutture grafo**

di Davide Marazza

Nella scorsa puntata abbiamo presentato le strutture grafo. Ricordiamo brevemente che un grafo è una struttura costituita da nodi che possono essere collegati tra loro da lati orientati. Vi avevamo presentato la prima parte del programma che gestisce tali strutture; mancava tuttavia la parte d'implementazione del modulo *Grafi* che contiene tutte le procedure fondamentali per la gestione di tali strutture e ovviamente per la compilazione del programma completo. Come prima cosa vediamo come sono state organizzate le variabili atte a contenere la struttura del grafo.

È opportuno, per la corretta comprensione di tale argomento, che si abbia ben chiaro com'è fatto un grafo; sarà utile quindi provare a disegnarne uno su un foglio. Procediamo come segue: disegnate un certo numero di punti sufficientemente distanti tra loro, numerateli progressivamente a partire da uno e poi collegateli a piacere con delle frecce, magari assegnando a tali frecce una distanza: avete appena disegnato un grafo.

Com'è organizzato il grafo

Ogni nodo del grafo è rappresentato da una variabile di tipo *NodoGrafo*, come si può vedere dal modulo di definizione *Grafi* pubblicato nella scorsa puntata. Il Tipo *NodoGrafo* è un Record che contiene una variabile di tipo *Elemento* (a sua volta un record) contenente il numero del nodo, e due puntatori di cui chiariamo subito lo scopo. Il primo, *ProssimoNodo*, punta al nodo successivo del grafo (in ordine sequenziale), in modo da collegare insieme tutta la struttura. Il secondo, *Adiacenti*, punta a una lista che contiene informazioni circa i nodi che nel grafo (da noi disegnato) sono collegati al nodo che stiamo considerando. La lista dei nodi adiacenti (ovvero collegati al nodo in considerazione) è semplicemente una lista sequenziale, argomento che è stato già trattato nelle puntate precedenti, e che quindi non riprendiamo in esame. Il puntatore a *ProssimoNodo* punterà a *NIL* quando si sarà giunti all'ultimo nodo contenuto nel

grafo. Per maggior chiarezza, facciamo uno schema di come si presentano i dati, utilizzando un simbolismo che speriamo risulti sufficientemente chiaro.

STRUTTURA GRAFO

NODO NODI COLLEGATI

```
NodoGrafo ---> Adiacente -> Adiacente ... -> NIL
|
NodoGrafo ---> ... -> NIL
|
... ---> ...
|
NIL
```

A questo punto, dovrete aver intuito il modo con cui si sono organizzati i dati allo scopo di ottenere una struttura grafo; tuttavia, ci teniamo a sottolineare che questa non è affatto l'unica maniera di organizzare una struttura grafo, ma è possibile immaginarne molte altre, che lasciamo alla vostra fantasia. Il nostro esempio è comunque utile per comprendere i meccanismi legati all'utilizzo dei puntatori, delle variabili Record e di moltissimi altri argomenti.

Un'altra struttura che viene spesso utilizzata è quella che contiene i percorsi possibili tra un nodo e un altro. Questa struttura può essere vista come tante liste (ognuna delle quali contiene un percorso possibile) collegate tra loro mediante un puntatore situato nell'elemento di testa di ogni lista; ovviamente, l'ultima lista avrà tale puntatore fissato a *NIL*.

I tipi di dati che interessano questa struttura sono: *TestaLista*, che punta al primo elemento della lista, *NodoLista*, che realizza tale primo elemento, ed *Elem*, che realizza tutti gli elementi successivi; volendo rifarsi a uno schema simile a quello visto in precedenza, avremo:

LISTA DI PERCORSI

```
TestaLista
|
```



```

NodoLista ---> Elem -> Elem .... -> NIL
|
NodoLista ---> ... -> NIL
|
... ---> ...
|
NIL

```

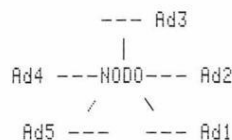
Le procedure del modulo Grafi

La prima procedura di cui ci occupiamo è Raggiungi, il cui scopo è verificare se da un certo nodo (NodoA) sia possibile o meno raggiungere, nel grafo, un altro nodo (NodoB). La procedura utilizza a sua volta un'altra procedura definita al suo interno, Raggiungibili. L'elemento principale di questa procedura è MiaLista, una lista sequenziale di elementi che viene riempita nel modo che ora vedremo. Inizialmente, viene aggiunto in MiaLista il NodoA. Successivamente, viene scandita la lista dei nodi adiacenti al NodoA e memorizzata nella struttura del grafo, allo scopo di controllare se il NodoB è tra quelli, nel qual caso la procedura dà un risultato positivo (TRUE) per indicare che il NodoA e il NodoB sono effettivamente raggiungibili. Se il NodoB, invece, non appartiene all'insieme dei nodi adiacenti al NodoA, allora si procede in modo ricorsivo; questo punto è abbastanza delicato e richiede un certo sforzo per essere compreso correttamente. Per ognuno dei nodi adiacenti al NodoA si procede nel seguente modo: viene richiamata la procedura Raggiungibili con i parametri: NodoA, il suddetto nodo adiacente; NodoB, il nodo a cui si vuole arrivare (lo stesso della prima chiamata). In questo modo, ci si ripresenta lo stesso problema di partenza, affrontato però da una diversa posizione e precisamente essendoci spostati su uno dei nodi adiacenti a quello da cui eravamo partiti. A questo punto la storiella di prima si ripete; se il NodoB è tra i nodi adiacenti a quello dove siamo ok, altrimenti si passa a uno di quelli adiacenti (rispetto al nodo dove siamo arrivati, cioè il corrente NodoA) e così via. Una volta esauriti tutti quelli adiacenti e avuto un responso negativo, avremo la certezza che NodoB è irraggiungibile.

Veniamo adesso a chiarire lo scopo di MiaLista che finora, per semplicità, è stata ignorata, ma la cui presenza è fondamentale. Infatti, può accadere quanto segue: se partiamo dal NodoA (quello originale) e ci spostiamo poi avanti, per esempio su uno degli adiacenti, può accadere che tra i nodi adiacenti a quest'ultimo ci sia il NodoA. Una precisazione: ricordiamoci che il grafo è ORIENTATO e il fatto che da 1 si possa andare a 2 non vuol dire affatto il contrario, a meno che non sia presente l'opportuno collegamento direzionato. Comunque, se dovesse accadere che il nodo da cui siamo partiti ricapiti tra quelli adiacenti al nodo in cui siamo, anche questo verrebbe elaborato come tutti gli altri. Ma attenzione, perché in questo modo siamo tornati da capo. Continueremmo a girare indefinitamente in questo anello senza mai fermarci. Ovviamente, volendo evitare che questo accada, è stata inserita MiaLista che a mano a mano che ci spostiamo dal nodo di partenza prende nota dei nodi calpestati (se così possiamo dire) e impedisce alla procedura di ripassarvi in iterazioni successive durante lo svolgersi della ricorsività. A questo punto, quindi, l'unico modo è che prima o poi il NodoB diventi il nodo adiacente al nodo dove siamo arrivati nel percorso o che non ci siano più nodi adiacenti, o per ultimo, che ce ne siano, ma che siano nodi già presenti nel percorso stesso (e che per quanto detto prima dovranno essere scartati).

La procedura in assoluto più complicata da capire è

EstraiIPercorsi, che ha lo scopo di estrarre dal grafo tutti i percorsi distinti che è possibile fare partendo da un nodo per raggiungerne un altro e di memorizzarli in una sequenza di liste come abbiamo mostrato all'inizio di questo stesso articolo. Questa procedura si avvale di alcune sottoprocedure definite al suo interno e che analizziamo qui di seguito una alla volta. Per prima cosa diamo un'occhiata al corpo principale di EstraiIPercorsi. All'inizio viene chiamata la procedura GeneraUnCammino, che ha come parametri due nodi (ovviamente) e una lista. La procedura, nel caso in cui i due nodi siano raggiungibili, mette nella lista la sequenza di nodi da attraversare per arrivare dal nodo di partenza a quello destinazione. Resta da chiarire in che modo venga scelto il percorso, vediamo. Occorre prendere in considerazione anche la procedura Prossimo che qui viene chiamata in causa. Cerchiamo di semplificare al massimo il discorso; supponiamo di essere in un determinato nodo; questo nodo avrà dei nodi adiacenti, che saranno in qualche modo, in un certo ordine sequenziale, memorizzati nella lista di nodi adiacenti al nodo in cui ci troviamo. Bene, supponiamo che tale ordinamento coincida col muoversi in senso antiorario per esempio:



che corrisponde a:

```
NODO ---> Ad1 -> Ad2 -> Ad3 -> Ad4 -> Ad5 - |
```

Ora analizziamo i parametri passati a Prossimo:

- Grafo è la struttura del grafo.
- Nodo è il nodo in cui siamo (il suo numero per la precisione).
- Precedente è un numero che indica quale adiacente dobbiamo prendere; se tale numero è negativo indica che dobbiamo prendere il primo adiacente, altrimenti quello nella posizione prossimo + 1.

Quindi, questa procedura serve per prendere il nodo adiacente dopo quello che avevamo in precedenza preso, seguendo l'ordine sequenziale con cui tali nodi adiacenti sono memorizzati nella struttura grafo.

Torniamo alla procedura GeneraUnCammino. Essendo inizialmente Prossimo negativo, la prima chiamata a Prossimo ci dà il primo (nel senso detto prima) nodo adiacente a quello in cui siamo; questo nodo è detto Corrente. A questo punto viene messo nella lista il nodo in cui ci troviamo e si passa a controllare se da Corrente si può raggiungere il nodo Destinazione. Facciamo a questo punto un passo in più. La lista di cui abbiamo parlato finora in realtà non è detto che sia una lista vuota, è possibile che per qualche motivo contenga già un certo percorso e che noi ci troviamo proprio sull'ultimo dei nodi di questo percorso. Allora, nell'andare a vedere se da Corrente si può raggiungere Destinazione dovremmo (per quanto già detto prima) porre attenzione a non calpestare nodi che sono già presenti nel percorso, altrimenti ricadremmo in cammini ciclici. Per questo motivo, quando viene chiamato Raggiungi viene anche passata la lista contenente il percorso fatto finora proprio per evitare questo problema.

Benissimo, ora procediamo. Se da Corrente si può raggiungere Destinazione, allora si richiamerà ricorsivamente GeneraUnCammino con Corrente come nodo iniziale e Destinazio-

ne come nodo destinazione; in questo modo, un nodo alla volta la lista viene generata, prendendo sempre a ogni bivio, come abbiamo già detto, l'adiacente che è il primo nell'ordine sequenziale a verificare le nostre richieste di raggiungibilità.

A questo punto, abbiamo in mano il primo cammino che viene messo nella lista dei cammini possibili. Il problema è tirare fuori tutti gli altri senza ovviamente creare dei doppioni indesiderati. Si procede nel modo seguente: ci si mette nel penultimo nodo del cammino che è stato appena generato, e si considera il fatto che si è già utilizzato il NodoB come adiacente (essendo l'ultimo nodo del percorso è per forza il nodo destinazione); per render conto di ciò, Precedente viene appunto a valere NodoB. Ora se esiste il modo di raggiungere NodoB passando da un altro Adiacente vuol dire che esiste un altro percorso che a partire dal nodo in cui siamo (il penultimo del percorso precedente) ci porta al NodoB. Consideriamo dunque come lista del percorso il percorso precedente troncato al penultimo nodo (cfr. Taglia(Cammino,fin)), si richiama GeneraUnPercorso che svolgerà la sua funzione come detto prima. Può succedere invece che non vi sia nessun nodo adiacente da cui si arriva al NodoB e allora ci si sposta indietro nel percorso di un altro nodo e si riprova a considerare gli adiacenti successivi a quello che era già stato preso nel percorso di prima. Se si arriva al NodoB si procede come prima, altrimenti si arretra ancora. Alla fine, si arriva a non poter più arretrare e ciò indica che tutti i percorsi sono

stati esaminati e si restituisce la lista dei percorsi che sono stati generati.

A questo punto, le procedure più complicate sono state esaminate sufficientemente e le altre sono lasciate a voi come ulteriore chiarimento di quanto fin qui appreso. Possiamo quindi considerare concluso anche quest'ultimo articolo con il quale si conclude il nostro piccolo corso sul *Modula 2*. Prima di salutarci, è opportuno però spendere due parole sul linguaggio e su quanto è stato fatto finora. Gli argomenti che abbiamo trattato evidentemente sono solo le fondamenta del linguaggio, la sua struttura di base. Non si è mai visto niente che riguardasse grafica suono o altre applicazioni strettamente specialistiche. Questo per ovvi motivi di spazio e tempo. Tenete quindi presente che il *Modula 2* non ha un'immagine tetra e scura, adatta solo a programmatori folli. Tale linguaggio possiede invece grandi capacità anche in campo grafico e sonoro e che (per esempio per quanto riguarda il *BenchMark*) esistono delle *simplified library* che rendono molto semplici le operazioni ai programmatori; queste librerie consentono di svolgere operazioni grafiche con la stessa facilità di un normale Basic, ma con la potenza e la flessibilità che solo un linguaggio quale il C può permettere.

Ora è venuto veramente il momento di salutarci. Il corso di *Modula 2* si conclude qui. Vi ringrazio per l'attenzione che avete dedicato a questa serie di articoli.

```
IMPLEMENTATION MODULE Grafi;
```

```
FROM InOut IMPORT WriteString,WriteInt,WriteLn,Read;
FROM Storage IMPORT ALLOCATE,DEALLOCATE;
FROM FileSystem IMPORT Lookup,Close,ReadWord,WriteWord,Response,
File;
FROM Strings IMPORT CompareString,ExtractSubString,StringLength,
Relation;
```

```
PROCEDURE StampaMessaggio(Err:CARDINAL);
(* stampa un messaggio di avvertimento *)
```

```
PROCEDURE StampaStringa(Stringa:ARRAY OF CHAR);
VAR c:CHAR;
Temp:ARRAY [1..3] OF CHAR;
BEGIN
  WriteLn;
  WriteString(Stringa);
  WriteLn;
  ExtractSubString(Temp,Stringa,0,3);
  IF NOT(CompareString(Temp,***)=equal) THEN
    Read(c);
  END;
END StampaStringa;
```

```
BEGIN
CASE Err OF
0: RETURN;|
1: StampaStringa("*** Il grafo non contiene nodi");|
2: StampaStringa("*** I due nodi non sono collegati");|
3: StampaStringa("*** Uno dei due nodi non e' nel grafo");|
4: StampaStringa("*** Non esistono percorsi");|
5: StampaStringa("*** Errore nel caricamento");|
6: StampaStringa("*** Errore nel salvataggio");|
7: StampaStringa("*** Memoria insufficiente");|
8: StampaStringa("*** Il nodo non e' nel grafo");|
9: StampaStringa("*** I due nodi sono coincidenti");|
END; (* CASE *)
END StampaMessaggio;
```

```
PROCEDURE CercaNodo (VAR Temp:PuntaNodo;Num:INTEGER):BOOLEAN;
BEGIN
IF Temp=NIL THEN
(*Grafo vuoto*)
RETURN(FALSE);
ELSE
WHILE Temp^.ProssimoNodo<>NIL DO
(*scende nella lista dei nodi*)
IF Temp^.Info.Numero=Num THEN
(*okay trovato nodo cercato*)
```

```
RETURN(TRUE);
ELSE
Temp:=Temp^.ProssimoNodo;
END;(*IF*)
END;(*WHILE*)
IF Temp^.Info.Numero=Num THEN
(*il nodo cercato e' l'ultimo in lista*);
RETURN(TRUE);
END;(*IF*)
IF Num=Bottom THEN
(*raggiunta fine lista come richiesto*)
RETURN(TRUE);
ELSE
(*nodo non trovato*)
RETURN(FALSE);
END;(*IF*)
END;(*IF*)
END CercaNodo;
```

```
PROCEDURE Stacca(VAR Temp:PuntaNodo; Nodo:INTEGER);
VAR Temp1,Temp2:PuntaAdiacente;
BEGIN
IF Temp=NIL THEN
(*vuoto*)
RETURN;
ELSIF Temp^.Adiacenti=NIL THEN
(*non ci sono adiacenti*)
RETURN;
ELSE
IF Temp^.Adiacenti^.Info.Numero=Nodo THEN
(*devo togliere il primo in lista*)
Temp1:=Temp^.Adiacenti;
Temp^.Adiacenti:=Temp1^.Prossimo;
DEALLOCATE(Temp1,SIZE(Temp1^));
ELSE
(*cerco tra i nodi collegati*)
Temp1:=Temp^.Adiacenti;
WHILE Temp1<>NIL DO
(*cerca il nodo da togliere*)
IF Temp1^.Info.Numero=Nodo THEN
(*trovato*)
Temp2^.Prossimo:=Temp1^.Prossimo;
DEALLOCATE(Temp1,SIZE(Temp1^));
RETURN;
ELSE
Temp2:=Temp1;
Temp1:=Temp1^.Prossimo;
END;(*IF*)
END;(*WHILE*)
END;(*IF*)
```



```

END; (*IF*)
END Stacca;

PROCEDURE AggiungiUnNodo (VAR Grafo:PuntaNodo);
VAR Nuovo,Temp:PuntaNodo;
BEGIN
  ALLOCATE(Nuovo,SIZE(Nuovo^));
  IF Nuovo=NIL THEN
    WriteString("Memoria insufficiente");
    RETURN;
  END; (* IF *)
  Nuovo^.Adiacenti:=NIL;
  Nuovo^.ProssimoNodo:=NIL;
  Temp:=Grafo;
  IF Temp=NIL THEN
    (*il grafo e' vuoto*)
    Grafo:=Nuovo;
    Nuovo^.Info.Numero:=1;
  ELSEIF CercaNodo(Temp,Bottom)=TRUE THEN
    (*ci sono dei nodi nella lista;
    mi porto sull'ultimo in lista e inserisco il nuovo nodo *)
    Nuovo^.Info.Numero:=Temp^.Info.Numero+1;
    Temp^.ProssimoNodo:=Nuovo;
  END; (*IF*)
END AggiungiUnNodo;

PROCEDURE TogliUnNodo(VAR Grafo:PuntaNodo;Nodo:INTEGER);
VAR Temp,Temp1:PuntaNodo;
BEGIN
  IF Grafo=NIL THEN
    (*Grafo vuoto*)
    StampaMessaggio(1);
    RETURN;
  ELSE
    (*il grafo contiene almeno un nodo*)
    Temp:=Grafo;
    (*cancella il nodo dalla lista*)
    IF Temp^.Info.Numero=Nodo THEN
      (*il nodo da togliere e' il primo*)
      Grafo:=Grafo^.ProssimoNodo;
      DEALLOCATE(Temp,SIZE(Temp^));
    ELSE
      WHILE Temp<>NIL DO
        IF Temp^.Info.Numero=Nodo THEN
          (*individuato nodo da togliere*)
          Temp1^.ProssimoNodo:=Temp^.ProssimoNodo;
          DEALLOCATE(Temp,SIZE(Temp^));
          Temp:=Grafo;
          WHILE Temp<>NIL DO
            Stacca(Temp,Nodo);
            Temp:=Temp^.ProssimoNodo;
          END; (*WHILE*)
          RETURN;
        ELSE
          (*scende lungo la lista*)
          Temp1:=Temp;
          Temp:=Temp^.ProssimoNodo;
        END; (*IF*)
      END; (*WHILE*)
      StampaMessaggio(8);
    END; (*IF*)
  END; (*IF*)
END TogliUnNodo;

PROCEDURE CollegaINodi (VAR Grafo:PuntaNodo;NodoA,NodoB:INTEGER);
VAR Temp,Temp1:PuntaNodo;
    Collegato,Collegamento:PuntaAdiacente;
BEGIN
  IF NodoA=NodoB THEN
    (*non si puo' collegare un nodo con se stesso*)
    StampaMessaggio(9);
    RETURN;
  ELSEIF Grafo=NIL THEN
    (*grafo vuoto*)
    StampaMessaggio(1);
    RETURN;
  ELSE
    (*il grafo contiene almeno un nodo*)
    Temp:=Grafo;
    Temp1:=Grafo;
    IF (CercaNodo(Temp,NodoA)=TRUE) AND (CercaNodo(Temp1,NodoB)=
    TRUE) THEN
      (*i due nodi sono in lista*)
      Stacca(Temp,NodoB);
      RETURN;
    ELSE
      (*uno dei due nodi non e' in lista*)
      StampaMessaggio(3);
      RETURN;
    END; (*IF*)
  END; (*IF*)
END CollegaINodi;

PROCEDURE UnisciINodi(VAR Grafo:PuntaNodo;NodoA,NodoB:INTEGER);
BEGIN
  CollegaINodi(Grafo,NodoA,NodoB);
  CollegaINodi(Grafo,NodoB,NodoA);
END UnisciINodi;

PROCEDURE SeparaINodi(VAR Grafo:PuntaNodo;NodoA,NodoB:INTEGER);
BEGIN
  ScollegaINodi(Grafo,NodoA,NodoB);
  ScollegaINodi(Grafo,NodoB,NodoA);
END SeparaINodi;

PROCEDURE Copia(UnaLista:PuntaElem):PuntaElem;
VAR Temp1,Temp2,NuovoElem,MiaLista:PuntaElem;
BEGIN
  IF UnaLista=NIL THEN
    (*la lista da copiare e' vuota*)
    RETURN(NIL);
  ELSE
    (*copia in MiaLista la lista passata*)
    ALLOCATE(MiaLista,SIZE(MiaLista^));
    Temp1:=UnaLista;
    Collegamento^.Info.Numero:=NodoB;
    Collegamento^.Prossimo:=NIL;
    Collegamento^.Info.Distanza:=0;
    Temp^.Adiacenti:=Collegamento;
    RETURN;
  ELSE
    (*ci sono gia' dei nodi collegati al nodoA*)
    WHILE Collegato^.Prossimo<>NIL DO
      (*si sposta alla fine dei collegamenti*)
      IF Collegato^.Info.Numero=NodoB THEN
        (*i due nodi sono gia' collegati*)
        RETURN;
      END; (*IF*)
      Collegato:=Collegato^.Prossimo;
    END; (*WHILE*)
    IF Collegato^.Info.Numero=NodoB THEN
      (*come sopra*)
      RETURN;
    END; (*IF*)
    ALLOCATE(Collegamento,SIZE(Collegamento^));
    Collegamento^.Info.Numero:=NodoB;
    Collegamento^.Prossimo:=NIL;
    Collegamento^.Info.Distanza:=0;
    Collegato^.Prossimo:=Collegamento;
  END; (*IF*)
ELSE
  (*uno dei due nodi non e' in lista*)
  StampaMessaggio(3);
  RETURN;
END; (*IF*)
END ScollegaINodi;

PROCEDURE ScollegaINodi (VAR Grafo:PuntaNodo;NodoA,NodoB:
INTEGER);
VAR Temp,Temp1:PuntaNodo;
BEGIN
  IF NodoA=NodoB THEN
    (*non si puo' scollegare un nodo con se stesso*)
    StampaMessaggio(9);
    RETURN;
  ELSEIF Grafo=NIL THEN
    (*Grafo vuoto*)
    StampaMessaggio(1);
    RETURN;
  ELSE
    (*il grafo contiene almeno un nodo*)
    Temp:=Grafo;
    Temp1:=Grafo;
    IF (CercaNodo(Temp,NodoA)=TRUE) AND (CercaNodo(Temp1,NodoB)=
    TRUE) THEN
      (*i due nodi sono in lista*)
      Stacca(Temp,NodoB);
      RETURN;
    ELSE
      (*uno dei due nodi non e' in lista*)
      StampaMessaggio(3);
      RETURN;
    END; (*IF*)
  END; (*IF*)
END ScollegaINodi;

PROCEDURE UnisciINodi(VAR Grafo:PuntaNodo;NodoA,NodoB:INTEGER);
BEGIN
  CollegaINodi(Grafo,NodoA,NodoB);
  CollegaINodi(Grafo,NodoB,NodoA);
END UnisciINodi;

PROCEDURE SeparaINodi(VAR Grafo:PuntaNodo;NodoA,NodoB:INTEGER);
BEGIN
  ScollegaINodi(Grafo,NodoA,NodoB);
  ScollegaINodi(Grafo,NodoB,NodoA);
END SeparaINodi;

PROCEDURE Copia(UnaLista:PuntaElem):PuntaElem;
VAR Temp1,Temp2,NuovoElem,MiaLista:PuntaElem;
BEGIN
  IF UnaLista=NIL THEN
    (*la lista da copiare e' vuota*)
    RETURN(NIL);
  ELSE
    (*copia in MiaLista la lista passata*)
    ALLOCATE(MiaLista,SIZE(MiaLista^));
    Temp1:=UnaLista;

```



```

Temp2:=MiaLista;
(*puntano rispettivamente a un elemento di UnaLista e di
MiaLista*)
Temp2^.Dato:=Temp1^.Dato;
Temp2^.Prox:=NIL;
WHILE Temp1^.Prox<>NIL DO
(*effettua la copia elemento per elemento fino a che UnaLista
non e' finita*)
Temp1:=Temp1^.Prox;
ALLOCATE(NuovoElem,SIZE(NuovoElem));
NuovoElem^.Dato:=Temp1^.Dato;
NuovoElem^.Prox:=NIL;
Temp2^.Prox:=NuovoElem;
Temp2:=NuovoElem;
END;(*WHILE*)
RETURN(MiaLista);
END;(*IF*)
END Copia;

```

```

PROCEDURE AttaccaAllaLista(VAR Lista:PuntaElem; Nodo:INTEGER);
(*aggiunge a Lista l'elemento Nodo*)
VAR Nuovo,Temp:PuntaElem;
BEGIN

```

```

Temp:=Lista;
ALLOCATE(Nuovo,SIZE(Nuovo));
Nuovo^.Dato:=Nodo;
Nuovo^.Prox:=NIL;
(*ha creato la parte di lista da aggiungere*)
IF Temp=NIL THEN
(*lista vuota*)
Lista:=Nuovo;
RETURN;
ELSE
(*lista contenente almeno un elemento*)
WHILE Temp^.Prox<>NIL DO
(*si sposta alla fine della lista*)
Temp:=Temp^.Prox;
END;(*WHILE*)
Temp^.Prox:=Nuovo;
END;(*IF*)
END AttaccaAllaLista;

```

```

PROCEDURE GiaInLista(Lista:PuntaElem;Nodo:INTEGER):BOOLEAN;
(*cerca se nodo e' contenuto in Lista e ritorna un booleano*)
VAR Temp:PuntaElem;
BEGIN

```

```

IF Lista=NIL THEN
(*lista vuota*)
RETURN(FALSE);
ELSE
(*lista non vuota*)
Temp:=Lista;
WHILE Temp<>NIL DO
IF Temp^.Dato=Nodo THEN
RETURN(TRUE);
ELSE
Temp:=Temp^.Prox;
END;(*IF*)
END;(*WHILE*)
RETURN(FALSE);
END;(*IF*)
END GiaInLista;

```

```

PROCEDURE Raggiungi(Grafo:PuntaNodo; Lista:PuntaElem;
NodoA,NodoB:INTEGER):BOOLEAN;
VAR c:CHAR;
Ris:BOOLEAN;
MiaLista:PuntaElem;

```

```

PROCEDURE Raggiungibili(Grafo:PuntaNodo; NodoA,NodoB:INTEGER):BO
OLEAN;
(*controlla la raggiungibilita' tra due nodi attraverso un
percorso semplice che non attraversi nodi di MiaLista*)
VAR Temp,Temp1:PuntaNodo;
Colleg:PuntaAdiacente;
BEGIN(*Raggiungibili*)
IF Grafo=NIL THEN
(*grafo vuoto*)
RETURN(FALSE);
ELSIF NodoA=NodoB THEN
(*i due nodi coincidono*)
RETURN(TRUE);
ELSE
AttaccaAllaLista(MiaLista,NodoA);
(*inserisce in MiaLista il NodoA*)
Temp:=Grafo;
Temp1:=Grafo;

```

```

IF (CercaNodo(Temp,NodoA)=TRUE) AND (CercaNodo(Temp1,
NodoB)=TRUE) THEN
(*i due nodi sono in lista*)
Colleg:=Temp^.Adiacenti;
(*Colleg punta a un nodo adiacente al NodoA*)
IF Colleg=NIL THEN
(*il nodoA e' isolato*)
RETURN(FALSE);
ELSE
(*cerco NodoB tra i nodi adiacenti*)
WHILE Colleg<>NIL DO
IF Colleg^.Info.Numero=NodoB THEN
(*il NodoB e' adiacente*)
RETURN(TRUE);
ELSE
(*cerca ancora*)
Colleg:=Colleg^.Prossimo;
END;(*IF*)
END;(*WHILE*)
Colleg:=Temp^.Adiacenti;
(*il NodoB non e' tra gli adiacenti al NodoA, allora
applico ricorsivamente il meccanismo di cui sopra
considerando NodoA un adiacente al NodoA di partenza
NB:devo stare attento a non cadere in cicli->MiaLista
serve a questo, non si considerano tra gli adiacenti
i nodi contenuti in MiaLista*)
WHILE Colleg<>NIL DO
IF GiaInLista(MiaLista,Colleg^.Info.Numero)=FALSE
THEN
(*evita i cicli infiniti nei grafi ciclici*)
(*applico la ricorsivita'*)
IF Raggiungibili(Grafo,Colleg^.Info.Numero,NodoB)=
TRUE THEN
RETURN(TRUE);
END;(*IF*)
END;(*IF*)
Colleg:=Colleg^.Prossimo;
END;(*WHILE*)
RETURN(FALSE);
END;(*IF*)
ELSE
(*uno o entrambi i nodi non sono in lista*)
RETURN(FALSE);
END;(*IF*)
END Raggiungibili;

```

```

BEGIN
MiaLista:=Copia(Lista);
(*Lista puo' contenere, per particolari scopi, degli eventuali
Nodi*)
Ris:=Raggiungibili(Grafo,NodoA,NodoB);
DEALLOCATE(MiaLista,SIZE(MiaLista));
RETURN(Ris);
END Raggiungi;

```

```

PROCEDURE DistanziaINodi (VAR Grafo:PuntaNodo;NodoA,NodoB,
Lunghezza:INTEGER);
VAR Temp,Temp1:PuntaNodo;
Colleg:PuntaAdiacente;

```

```

BEGIN
IF Grafo=NIL THEN
(*grafo vuoto*)
StampaMessaggio(1);
RETURN;
ELSE
(*grafo contiene nodi*)
Temp:=Grafo;
Temp1:=Temp;
IF (CercaNodo(Temp,NodoA)=TRUE) AND (CercaNodo(Temp1,NodoB)=
TRUE) THEN
(*i due nodi sono nel grafo*)
Colleg:=Temp^.Adiacenti;
(*controlla se nodoB e' collegato*)
WHILE Colleg<>NIL DO
IF Colleg^.Info.Numero=NodoB THEN
(*trovato*)
Colleg^.Info.Distanza:=Lunghezza;
RETURN;
ELSE
(*avanza nella lista degli adiacenti*)
Colleg:=Colleg^.Prossimo;
END;(*IF*)
END;(*WHILE*)
(*nodo non collegato*)
StampaMessaggio(2);

```



```

RETURN;
ELSE
  (*i due nodi non sono nel grafo*)
  StampaMessaggio(3);
  RETURN;
END;(*IF*)
END DistanziaINodi;

PROCEDURE PremiUnTasto;
VAR c:CHAR;
BEGIN
  WriteLn;
  WriteString("Premere un tasto per continuare");
  WriteLn;
  Read(c);
END PremiUnTasto;

```

```

PROCEDURE Pausa(VAR Numero:CARDINAL);
BEGIN
  IF Numero>=22 THEN
    PremiUnTasto;
    Numero:=0;
  END;(*IF*)
END Pausa;

```

```

PROCEDURE Vedo (Grafo:PuntaNodo);
VAR Temp:PuntaNodo;
    Temp1:PuntaAdiacente;
    Cont:CARDINAL;
BEGIN
  Cont:=0;
  (*conta le linee stampate*)
  IF Grafo=NIL THEN
    (*grafo vuoto*)
    StampaMessaggio(1);
    RETURN;
  ELSE
    WriteLn;
    WriteString("STRUTTURA GRAFO : ");
    WriteLn;
    Temp:=Grafo;
    (*stampa la lista dei nodi*)
    WHILE Temp<>NIL DO
      WriteString("NODO ");
      WriteInt(Temp^.Info.Numero,4);
      Cont:=Cont+1;
      Temp1:=Temp^.Adiacenti;
      (*stampa nodi collegati*)
      IF Temp1=NIL THEN
        (*nessun nodo collegato*)
        WriteString("..nessun adiacente");
      ELSE
        WriteString("->");
        WHILE Temp1<>NIL DO
          (*stampa la lista di nodi adiacenti*)
          WriteInt(Temp1^.Info.Numero,1);
          WriteString(" ");
          WriteInt(Temp1^.Info.Distanza,1);
          WriteString(" ");
          Temp1:=Temp1^.Prossimo;
        END;(*WHILE*)
      END;(*IF*)
      WriteLn;
      Pausa(Cont);
      Temp:=Temp^.ProssimoNodo;
    END;(*WHILE*)
  END;(*IF*)
END Vedo;

```

```

PROCEDURE Stampa (Lista:PuntaElem);
(*stampa una lista di numeri*)
VAR Temp:PuntaElem;
BEGIN
  Temp:=Lista;
  WriteLn;
  WriteString("LISTA NODI: ");
  WHILE Temp<>NIL DO
    WriteInt(Temp^.Dato,4);
    Temp:=Temp^.Prox;
  END;(*WHILE*)
END Stampa;

```

```

PROCEDURE StampaLista(Titolo:ARRAY OF CHAR; ListaCamm:TestaLista;
  Par:INTEGER);
(*stampa una lista di percorsi*)

```

```

VAR Temp:TestaLista;
    Cont:CARDINAL;
BEGIN
  WriteLn;
  WriteString(Titolo);
  WriteLn;
  Temp:=ListaCamm;
  Cont:=0;
  WHILE Temp<>NIL DO
    (*stampa per ogni percorso di ListaCamm la corrispondente lista
    dei nodi fermandosi dopo aver stampato un numero di percorsi
    pari a 22*)
    IF Par<0 THEN
      Stampa(Temp^.ListaAdiacente);
      Cont:=Cont+1;
      IF Temp^.Prossimo<>NIL THEN
        Pausa(Cont);
      END;
    ELSEIF Temp^.Distanza=Par THEN
      Stampa(Temp^.ListaAdiacente);
      Cont:=Cont+1;
      IF Temp^.Prossimo<>NIL THEN
        Pausa(Cont);
      END;
    END;(*IF*)
    Temp:=Temp^.Prossimo;
  END;(*WHILE*)
END StampaLista;

```

```

PROCEDURE EstraiIPercorsi(Grafo:PuntaNodo; NodoA,NodoB:INTEGER);
TestaLista;

```

```

PROCEDURE Prossimo(Grafo:PuntaNodo; Nodo,Precedente:INTEGER);
INTEGER;
(*prende il nodo adiacente a Nodo successivo a Precedente
se Precedente e' negativo prende il primo*)
VAR Temp:PuntaNodo;
    Temp1:PuntaAdiacente;
    Ris:BOOLEAN;
BEGIN
  Temp:=Grafo;
  Ris:=CercaNodo(Temp,Nodo);
  (*Temp ora punta a Nodo*)
  Temp1:=Temp^.Adiacenti;
  IF Temp1=NIL THEN
    (*Nodo non ha adiacenti-> risultato negativo*)
    RETURN(-1);
  ELSEIF Precedente<0 THEN
    (*restituire numero primo nodo adiacente*)
    RETURN(Temp1^.Info.Numero);
  ELSE
    WHILE Temp1^.Prossimo<>NIL DO
      (*cerca Precedente tra gli adiacenti*)
      IF Temp1^.Info.Numero=Precedente THEN
        Temp1:=Temp1^.Prossimo;
        (*Temp1 si posiziona sull'adiacente successivo a
        Precedente*)
        IF Temp1=NIL THEN
          RETURN(-1);
          (*non ci sono piu' adiacenti*)
        ELSE
          RETURN(Temp1^.Info.Numero);
        END;(*IF*)
      ELSE
        Temp1:=Temp1^.Prossimo;
      END;(*IF*)
    END;(*WHILE*)
    RETURN(-1);
  END;(*IF*)
END Prossimo;

```

```

PROCEDURE CalcolaDistanza(Grafo:PuntaNodo;VAR ListaPercorsi;
  TestaLista);
(*calcola la distanza lungo tutti i cammini di ListaPercorsi*)
VAR Temp:TestaLista;
    Temp1:PuntaElem;
    Distanza,Nodo1,Nodo2:INTEGER;

```

```

PROCEDURE Cammino (Grafo:PuntaNodo;Nodo1,Nodo2:INTEGER);
INTEGER;
(*calcola la distanza tra due nodi che sicuramente sono
collegati nel grafo*)
VAR Temp:PuntaNodo;
    Temp1:PuntaAdiacente;
    Ris:BOOLEAN;
BEGIN
  Temp:=Grafo;

```



```

Ris:=CercaNodo(Temp,Nodo1);
(*Temp si e' posizionato su Nodo1*)
Temp1:=Temp^.Adiacenti;
WHILE Temp1^.Info.Numero<>Nodo2 DO
  Temp1:=Temp1^.Prossimo;
END;(*WHILE*)
RETURN(Temp1^.Info.Distanza);
END Cammino;

BEGIN(*CalcolaDistanza*)
IF ListaPercorsi=NIL THEN
  RETURN;
ELSE
  Temp:=ListaPercorsi;
  WHILE Temp<>NIL DO
    (*calcola la distanza lungo il percorso puntato da Temp*)
    Distanza:=0;
    Temp1:=Temp^.ListaAdiacenti;
    (*punta al primo nodo del percorso puntato da Temp*)
    Nodo1:=Temp1^.Dato;
    Temp1:=Temp1^.Prox;
    (*punta al secondo nodo...*)
    WHILE Temp1<>NIL DO
      (*controlla se esiste il secondo, in tal caso somma la
      distanza tra i due nodi a quella parziale*)
      Nodo2:=Temp1^.Dato;
      Distanza:=Distanza+Cammino(Grafo,Nodo1,Nodo2);
      Nodo1:=Nodo2;
      Temp1:=Temp1^.Prox;
      (*avanza lungo il percorso ponendo come Nodo1 quello
      che prima era il Nodo2*)
    END;(*WHILE*)
    Temp^.Distanza:=Distanza;
    Temp:=Temp^.Prossimo;
    (*si posiziona sul percorso successivo*)
  END;(*WHILE*)
END;(*IF*)
END CalcolaDistanza;

PROCEDURE GeneraUnCammino(VAR Lista:PuntaElem;
  Adiacente, Destinazione:INTEGER)
  :BOOLEAN;
(*Genera la coda di lista a partire dal nodo adiacente fino ad
arrivare al nodo destinazione e restituisce FALSE in caso di
impossibilita' nel creare -> ciclicita'*)
VAR Corrente,Precedente:INTEGER;
Lista1:PuntaElem;
BEGIN
IF Adiacente=Destinazione THEN
  (*la lista finisce qui, cioe' sono arrivato a Destinazione*)
  AttaccaAllaLista(Lista, Destinazione);
  RETURN(TRUE);
ELSE
IF Raggiungi(Grafo,Lista,Adiacente, Destinazione)=FALSE THEN
  (*impossibile raggiungere Destinazione*)
  RETURN(FALSE);
ELSIF GiaInLista(Lista,Adiacente)=TRUE THEN
  (*Cammino ciclico e percio' lo si esclude*)
  RETURN(FALSE);
ELSE
  (*il cammino comprende piu' di due nodi*)
  Precedente:=-1;
  (*cfr. Prossimo*)
  LOOP
    Corrente:=Prossimo(Grafo,Adiacente,Precedente);
    (*Corrente e' l'ultimo nodo considerato nel cammino*)
    Lista1:=Copia(Lista);
    (*Costruisco una copia di Lista che mi servira' tra
    poco per controllare da che parte dirigermi nel
    cammino*)
    AttaccaAllaLista(Lista1,Adiacente);
    (*suppongo di potere passare dall'adiacente che sto
    considerando per raggiungere la destinazione*)
    IF (Raggiungi(Grafo,Lista1,Corrente, Destinazione)=TRUE)
    AND (GiaInLista(Lista,Corrente)=FALSE) THEN
      (*la supposizione era giusta e aggiorni anche la
      lista effettiva del cammino*)
      AttaccaAllaLista(Lista,Adiacente);
      RETURN(GeneraUnCammino(Lista,Corrente, Destinazione));
      (*a questo punto ricorsivamente considero come nodo
      di partenza il Corrente e applico lo stesso
      algoritmo di prima utilizzando come Lista il
      cammino finora generato*)
    ELSE
      (*la supposizione era errata: prova con un altro
      adiacente*)
      Precedente:=Corrente;
    END;
  END LOOP;
END;

```

```

END;(*IF*)
END;(*LOOP*)
END;(*IF*)
END;(*IF*)
END GeneraUnCammino;

PROCEDURE Link(ListaCammini:TestaLista; UnaLista:PuntaElem):
  TestaLista;
(*Attacca alle liste dei cammini generati UnaLista (un cammino)
e restituisce il puntatore all'intera struttura*)
VAR Temp,Nuovo:TestaLista;
MiaLista:PuntaElem;
BEGIN
IF UnaLista=NIL THEN
  (*lista vuota*)
  RETURN(ListaCammini);
END;(*IF*)
MiaLista:=Copia(UnaLista);
IF ListaCammini=NIL THEN
  (*e' il primo cammino della struttura ad essere inserito*)
  ALLOCATE(ListaCammini,SIZE(ListaCammini));
  ListaCammini^.ListaAdiacente:=MiaLista;
  ListaCammini^.Prossimo:=NIL;
ELSE
  (*ci sono gia' degli altri percorsi; mi sposto sull'ultimo
  e aggiungo poi quello nuovo*)
  Temp:=ListaCammini;
  WHILE Temp^.Prossimo<>NIL DO
    Temp:=Temp^.Prossimo;
  END;(*WHILE*)
  ALLOCATE(Nuovo,SIZE(Nuovo));
  Nuovo^.ListaAdiacente:=MiaLista;
  Nuovo^.Prossimo:=NIL;
  Temp^.Prossimo:=Nuovo;
END;(*IF*)
RETURN(ListaCammini);
END Link;

PROCEDURE Lunghezza(Lista:PuntaElem):INTEGER;
(*restituisce il numero dei nodi da attraversare lungo il
cammino Lista*)
VAR Temp:PuntaElem;
Cont:INTEGER;
BEGIN
  Cont:=0;
  Temp:=Lista;
  WHILE Temp<>NIL DO
    Cont:=Cont+1;
    Temp:=Temp^.Prox;
  END;
  RETURN(Cont);
END Lunghezza;

PROCEDURE Estrai(Lista:PuntaElem; Numero:INTEGER):INTEGER;
(*restituisce il numero del nodo nella posizione Numero della
Lista*)
VAR Temp:PuntaElem;
Cont:INTEGER;
BEGIN
  Cont:=1;
  Temp:=Lista;
  WHILE Cont<>Numero DO
    Cont:=Cont+1;
    Temp:=Temp^.Prox;
  END;(*WHILE*)
  RETURN(Temp^.Dato);
END Estrai;

PROCEDURE Taglia(Lista:PuntaElem; Numero:INTEGER):PuntaElem;
(*restituisce il puntatore alla parte di Lista contenente i
primi Numero nodi*)
VAR Temp,Temp1:PuntaElem;
Cont:INTEGER;
BEGIN
  Cont:=1;
  Temp:=Lista;
  WHILE Cont<>Numero DO
    Cont:=Cont+1;
    Temp:=Temp^.Prox;
  END;(*WHILE*)
  Temp1:=Temp^.Prox;
  IF Temp1=NIL THEN
    (*devo prendere l'intera lista*)
  ELSE
    (*elimino quella parte di lista che non serve piu'*)
    Temp^.Prox:=NIL;
    DEALLOCATE(Temp1,SIZE(Temp1));
  END;
END;

```



```

END; (* IF *)
RETURN(Lista);
END Taglia;

VAR Cammino:PuntaElem;
ListaCammini:TestaLista;
Fin,Precedente,Adiacente:INTEGER;
Generato:BOOLEAN;
BEGIN
Cammino:=NIL;
ListaCammini:=NIL;
(* genera il primo percorso *)
IF GeneraUnCammino(Cammino,NodoA,NodoB)=FALSE THEN
  (*non e' possibile alcun percorso*)
  StampaMessaggio(4);
  RETURN(NIL);
ELSE
  (*e' stato generato un percorso dalla precedente chiamata
  dell'if ed ora viene linkato alla lista dei percorsi*)
  ListaCammini:=Link(ListaCammini,Cammino);
END; (* IF *)
(*continua a generare cammini*)
Precedente:=NodoB;
LOOP
Fin:=Lunghezza(Cammino)-1;
(*fin e' un numero intero che indica la penultima posizione in
Cammino che e' l'ultimo cammino generato; serve per tentare
di creare dei percorsi partendo dal penultimo nodo di
Cammino diversi da quelli fin ora generati*)
Generato:=FALSE;
WHILE Generato=FALSE DO
  Adiacente:=Prossimo(Grafo,Estrai(Cammino,Fin),Precedente);
  (* considera il nodo adiacente a quello nella posizione in-
  dicata da fin nel Cammino, successivo all'adiacente pre-
  cedente che e' il nodo nella posizione fin+1 in Cammino *)
  IF Adiacente>0 THEN
    (* esiste un tale adiacente *)
    Cammino:=Taglia(Cammino,Fin);
    IF GeneraUnCammino(Cammino,Adiacente,NodoB)=TRUE THEN
      (*il cammino non e' mai stato generato in precedenza*)
      ListaCammini:=Link(ListaCammini,Cammino);
      Generato:=TRUE;
      Precedente:=NodoB;
    ELSE
      (*e' impossibile generare un cammino dal nodo adiacente
      considerato*)
      Precedente:=Adiacente;
    END; (* IF *)
  ELSE
    (*sono finiti i nodi adiacenti al nodo indicato da fin
    in Cammino*)
    Fin:=Fin-1;
    IF Fin<1 THEN
      (*se fin esce dal cammino => fine operazione *)
      CalcolaDistanza(Grafo,ListaCammini);
      RETURN(ListaCammini);
    ELSE
      (* si sposta sul nodo precedente a quello indicato da
      fin e procede come sopra *)
      Precedente:=Estrai(Cammino,Fin+1);
      (*prendero' il nodo adiacente a quello indicato da fin
      successivo all'adiacente precedente che e' quello nella
      posizione fin+1; questo per evitare inutili
      ripetizioni nella generazione dei cammini*)
    END; (* IF *)
  END; (* IF *)
END; (* WHILE *)
END; (* LOOP *)
END EstraiIPercorsi;

PROCEDURE SalvaGrafo(Grafo:PuntaNodo; Nome:ARRAY OF CHAR);
Response;
VAR Temp:PuntaNodo;
FileOut:File;
Temp1:PuntaAdiacente;
BEGIN
IF (Grafo=NIL) OR (StringLength(Nome)=0) THEN
  (*non c'e' niente da salvare*)
  RETURN(notdone);
ELSE
  Lookup(FileOut,Nome,TRUE);
  (* tenta di aprire il file Nome *)
  IF FileOut.res<>done THEN
    (*insuccesso*)
    RETURN(FileOut.res);
  END; (* IF *)
  Temp:=Grafo;

```

```

(* genera un nuovo file di output e salva su disco le
informazioni fondamentali sulla struttura del grafo *)
WHILE Temp<>NIL DO
  WriteWord(FileOut,Temp^.Info.Numero);
  Temp1:=Temp^.Adiacenti;
  WHILE Temp1<>NIL DO
    (*Scrive l'elenco dei nodi adiacenti e le distanze dalla
    radice*)
    WriteWord(FileOut,Temp1^.Info.Numero);
    WriteWord(FileOut,Temp1^.Info.Distanza);
    Temp1:=Temp1^.Prossimo;
  END; (* WHILE *)
  WriteWord(FileOut,-1);
  (* separatore nel file *)
  Temp:=Temp^.ProssimoNodo;
END; (* WHILE *)
Close(FileOut);
(*salvataggio terminato*)
END; (* IF *)
RETURN(done);
END SalvaGrafo;

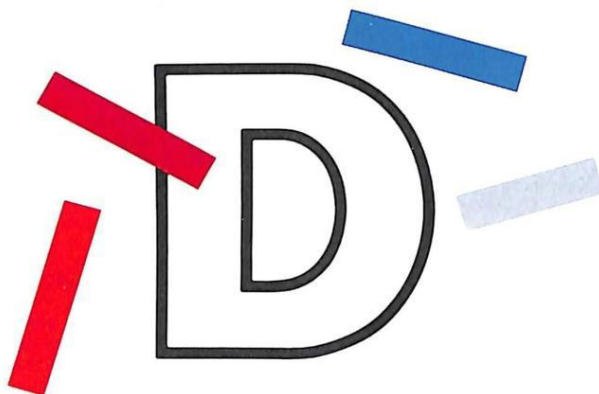
PROCEDURE CaricaGrafo(VAR Grafo:PuntaNodo; Nome:ARRAY OF CHAR);
Response;
(* tenta di caricare in memoria -> Grafo, il file Nome che DEVE
contenere la struttura di un grafo *)
VAR Temp,NuovoNodo:PuntaNodo;
FileIn:File;
Dato:INTEGER;
Temp1,NuovoAdiacente:PuntaAdiacente;
BEGIN
IF StringLength(Nome)=0 THEN
  (*nome illegale*)
  RETURN(notdone);
ELSE
  (*prova ad aprire il file di lettura*)
  Lookup(FileIn,Nome,FALSE);
  IF FileIn.res<>done THEN
    (*c'e' stato qualche problema*)
    RETURN(FileIn.res);
  ELSE
    Grafo:=NIL;
    ReadWord(FileIn,Dato);
    (*controlla che il file contenga qualcosa*)
    WHILE Dato<>-1 DO
      ALLOCATE(NuovoNodo,SIZE(NuovoNodo));
      (* aggiunge un nodo al Grafo *)
      NuovoNodo^.Info.Numero:=Dato;
      NuovoNodo^.ProssimoNodo:=NIL;
      NuovoNodo^.Adiacenti:=NIL;
      IF Grafo=NIL THEN
        (*primo nodo*)
        Grafo:=NuovoNodo;
      ELSE
        Temp^.ProssimoNodo:=NuovoNodo;
      END; (* IF *)
      Temp:=NuovoNodo;
      Temp1:=Temp^.Adiacenti;
      ReadWord(FileIn,Dato);
      (*collega gli adiacenti al nodo creato*)
      WHILE Dato<>-1 DO
        ALLOCATE(NuovoAdiacente,SIZE(NuovoAdiacente));
        NuovoAdiacente^.Prossimo:=NIL;
        NuovoAdiacente^.Info.Numero:=Dato;
        ReadWord(FileIn,Dato);
        NuovoAdiacente^.Info.Distanza:=Dato;
        IF Temp1=NIL THEN
          Temp^.Adiacenti:=NuovoAdiacente;
        ELSE
          Temp1^.Prossimo:=NuovoAdiacente;
        END; (* IF *)
        Temp1:=NuovoAdiacente;
        ReadWord(FileIn,Dato);
      END; (* WHILE *)
      (*legge prossimo nodo*)
      ReadWord(FileIn,Dato);
    END; (* WHILE *)
  END; (* IF *)
  Close(FileIn);
  RETURN(done);
END; (* IF *)
END CaricaGrafo;

END Grafi.

```


COMPUTER E DIDATTICA

ESPERIENZE DIDATTICHE A CONFRONTO



INTERVISTIAMO IL BTD

Questo mese vi proponiamo un'intervista (realizzata via modem) con Giorgio Banaudi, docente di Lettere in una scuola media di Genova e sysop del Bollettino sulle Tecnologie Didattiche, un BBS appartenente alla rete Fidonet.

D. Quando e come è nata l'idea di questo BBS e da quando esattamente è entrato ufficialmente in funzione?

R. Partiamo proprio dalle origini. La Biblioteca del Software Didattico (una realizzazione dell'Istituto per le Tecnologie Didattiche del CNR, con sede a Genova, ITD), nell'ambito dei suoi progetti di ricerca e di sperimentazione ha attivato, a partire dal settembre '90, un BBS denominato Bollettino sulle Tecnologie Didattiche (BTD), nato espressamente per verificare l'ipotesi che un sistema telematico amatoriale potesse inserirsi fattivamente come strumento e servizio nel campo della didattica. L'ITD è da diversi anni impegnato nel campo della ricerca didattica, con sperimentazioni di nuove tecnologie, addestramento di insegnanti e seminari divulgativi, ma questa occasione si configurava come una pista ancora inesplorata e promettente. Abbiamo poi cercato un software di gestione BBS e siamo partiti in configurazione minima: un BBS locale, senza altri collegamenti, con alcune aree tematiche e una discreta dotazione di materiale didattico proveniente direttamente dalla BSD. Il veloce diffondersi via modem dell'iniziativa ha portato ben presto a un'utenza giornaliera discreta, quantificabile sui dieci

utenti al giorno. Poi, nel maggio del 1991, abbiamo deciso di entrare nella grande rete Fidonet, partecipando direttamente ai dibattiti di alcune conferenze e divulgando le iniziative e il materiale di cui disponevamo.

D. Quali sono le sue finalità in generale?

R. Il BTD intende privilegiare particolarmente questi settori:

- uso di tecnologie informatiche nella didattica;
- consultazione e ricerca del software didattico disponibile in Italia;
- software e sperimentazioni rivolte al settore dei portatori di handicap;
- scambio d'informazioni inerenti a questi temi tra operatori;
- divulgazione delle attività svolte dalla BSD-ITD (seminari, convegni...).

L'accesso è libero e gratuito per chiunque si ritenga interessato, anche

Questa rubrica si basa su testimonianze di alunni ed insegnanti, che si interessano alla didattica su computer. I contributi editoriali sono grandemente apprezzati. Inviare eventuali materiali (articoli, foto, disegni, descrizioni di esperienze...) a:

COMMODORE GAZETTE
Comodore e didattica
Via Monte Napoleone, 9
20121 Milano

se ovviamente si cerca d'invogliare soprattutto gli insegnanti e i docenti, gli operatori didattici e chi fa sperimentazione nella scuola. Chiaramente anche gli studenti più motivati sono ben accolti per tessere un dialogo che non bada tanto all'autorevolezza fisica della persona, ma ai contenuti degli interventi da qualunque parte provengano.

D. E in particolare?

R. Ultimamente ci siamo accorti che i campi specifici sui quali insistiamo maggiormente sono soprattutto la funzione di concentrare e divulgare le esperienze di cui veniamo a conoscenza, fornire un punto di riferimento per quanto riguarda l'uso della "telematica povera" nel campo della scuola, e convogliare nuove informazioni, annunci e progetti legati a queste tematiche verso chi non avrebbe la possibilità di accedervi. Pensiamo per esempio all'enorme mole informativa che si può ricavare dalla frequentazione delle reti geografiche universitarie (Internet) e dalle competenze specifiche dei vari operatori del centro che quotidianamente sono implicati nel lavoro di ricerca.

D. Quali sono le sue caratteristiche tecniche?

R. Il BTD punta essenzialmente sui contenuti e le caratteristiche tecniche sono viste esclusivamente come un mezzo, non come un orpello da coltivare quasi con atteggiamento narcisistico. Per questo non si è dato un eccessivo rilievo alla cura dell'interfaccia utente che è e rimane quella classi-

ca del collaudato software *Opus 1.14*, che ha determinato un comodo standard per quanto riguarda l'uniformità dei menu sui vari BBS, in particolare nell'ambito Fidonet. La macchina utilizzata è un compatibile 386SX, con hard disk da 135 MB (ci stiamo comunque orientando verso una risistemazione e un ampliamento del sistema, soprattutto per quanto riguarda la capienza dell'hard disk). Abbiamo già provveduto ad adottare un modem veloce (v32bis) e abbiamo aggiornato il sistema operativo per utilizzare un semplice multitasking, in pratica facendo funzionare il BTd sotto *Windows*, considerato che fino al settembre '92 il BTd era monoutente e ogni intervento per la manutenzione causava la "chiusura" del sistema per lavori in corso. Ora invece queste pause forzate sono finalmente eliminate. Le modalità di accesso sono libere e gratuite. Chiediamo ai nostri utenti di fornirci alcuni essenziali dati anagrafici (il nominativo e un recapito telefonico, per garantire la loro e la nostra serietà) e dopo l'abilitazione al sistema (che avviene in pochissimi giorni) l'utente può collegarsi per un tempo limite di 30 minuti, più che sufficiente per svolgere tutte le funzioni richieste (in effetti, dai nostri rilevamenti, risulta che un utente medio rimane collegato per 10/15 minuti).

D. Com'è strutturato il BTd?

R. Il BTd mette liberamente a disposizione dei suoi utenti: MESSAGGI (conferenze locali su tematiche inerenti il centro ITD e BSD; conferenze in echo con la rete Fidonet con particolare riferimento ai settori umanistici); FILES (raccolta di documentazione su didattica, handicap, medicina, ampia collezione di software didattico PD in particolare per sistemi MS-DOS, demo di programmi educativi, testi umanistici, software PD rivolto a portatori di handicap); BBSD (consultazione on-line della Banca Dati della Biblioteca del Software Didattico che dal 1985 raccoglie e cataloga il software didattico italiano ed estero); BOLLETTINO SULLE ATTIVITÀ DELL'ISTITUTO TECNOLOGIE DIDATTICHE; consultazione del Calendario dei corsi e convegni (liberi e gratuiti) organizzati dall'ITD in collaborazione con IRRSAE Liguria; possibilità d'iscrizione e richiesta di consultazioni della BBSD.

D. Quante e quali sono le aree messaggi locali e generali?

R. Con l'ingresso nella rete Fidonet, scelta proprio per la sua capillare presenza e versatilità, abbiamo scelto con cura le aree che ci sembravano più in linea con i nostri obiettivi, in pratica chi si collega oggi al BTd può trovare le seguenti aree della rete Fidonet italiana: HUMAN.ITA (problemi socio-umanitari, handicap...), SALUTE.ITA (problemi di salute, medicina), LIBRI.ITA (discussioni sui libri, novità, recensioni), SCIENZA.ITA (argomenti vari di scienza), AI.ITA (discussioni sull'Intelligenza Artificiale), DEWDNEY.ITA (ricreazioni al calcolatore), MATE.ITA (la matematica e il mondo dei numeri), UNIVERSITA.ITA (l'università e i suoi studenti), SCUOLA.ITA (il mondo della scuola, didattica...), ECOLOGIA (l'ambiente, la sua difesa, i suoi problemi), PEACELINK.ITA (educazione alla pace e alla mondialità), CINEMA (film e dintorni), MUSICA (il mondo delle sette note).

D. E le aree file?

R. Per dare un'idea della struttura attuale delle aree file riportiamo l'elenco completo: DOCUMENTI e TESTI di INTERESSE GENERALE, ARTICOLI DI RIVISTE DIDATTICHE, PUBBLICAZIONI della BSD e ITD, FREE SOFTWARE DIDATTICO per MS-DOS, FREE SOFTWARE didattico per AMIGA, FREE SOFTWARE didattico per APPLE, FREE SOFTWARE didattico per MAC, FREE SOFTWARE didattico per altri computer, PROGRAMMI DI UTILITÀ VARIA, PROGRAMMI DI COMUNICAZIONE, TELEVIDEO - programmi distribuiti dalla RAI, AREA PER SCAMBIO PROGRAMMI PERSONALI TRA UTENTI, DOCUMENTAZIONE BSD-MEDICINA, DOCUMENTAZIONE BSD-HANDICAP, DEMO di programmi didattici, HANDICAP DIGEST, WINDOWS 3 PD, CLIP-ART Immagini, Art-lines, Figure, ED TECH - Tecnologia applicata all'educazione, OPUS - per mettere su un BBS..., KIDLINK & KIDS-92, File in arrivo da FIDONET - INBOUND AREA. E per incrementare ulteriormente il "magazzino" del BTd ogni tanto anche noi ci lanciamo in "cacce al tesoro telematico" per reperire nuovo materiale software, che generalmente rintracciamo sui grossi server americani, ai quali siamo collegati tramite la rete Internet.

D. Per chi si collega al BBS, è possibile accedere e consultare telematicamente la Biblioteca del Software Didattico?

R. L'idea di fornire uno strumento per consultare la ricca Biblioteca del Software ci era sembrata fin dall'inizio un elemento qualificante. Questa banca dati era già disponibile tramite Videotel (consultando la BDP, Banca Dati Pedagogica di Firenze), ma al momento è riservata agli utenti del gruppo Scuolatel, riservato a scuole e difficilmente accessibile anche dal singolo docente. Un'altra possibilità di consultarla in remoto è quella di accedere a un nodo della rete EARN, anche questa riservata a specialisti e a pochi fortunati utenti di Internet. Insomma, l'unica alternativa per la maggioranza delle persone era la consultazione diretta e personale nel nostro centro di Genova. Volevamo quindi fornire un semplice mezzo per dare almeno un'idea del materiale che si trova in questa ricca collezione di software educativo. A tale scopo, è stato preparato un semplice programma di visualizzazione dell'archivio della BSD che, pur restringendo l'analisi ad alcuni campi, permette un esauriente approccio all'archivio. Questa funzione viene offerta a tutti gli utenti abilitati del BTd. Ovviamente, si può solo visionare l'archivio, non accedere e/o prelevare i programmi stessi; tramite un menu si possono fare ricerche e selezioni incrociate sui principali campi in cui si articola l'archivio: Materia, Scuola, Computer, Argomento e, quindi, visionare le singole schede con le voci relative (con un discreto livello di approfondimento), oppure conteggiarle, scorrerle in formato tabellare e infine, se la selezione effettuata non è troppo estesa (ovvero, se non supera le 100 schede, il che equivarrebbe a un documento compresso di circa 50K), è possibile prelevare il file di testo con la descrizione completa dei programmi desiderati, compresi quei campi che per comodità non vengono mostrati nell'interrogazione on-line.

D. Com'è il traffico dei messaggi inerenti al mondo scolastico, dell'insegnamento, dello studio, della ricerca e dello scambio d'informazioni tra insegnanti?

R. Tra le aree locali attive solo sul BTd (che spesso sono il tallone d'Achille dei BBS) e quelle in rete Fidonet, condivise quindi da molti altri BBS, ci sono diversi argomenti che rivestono un interesse significativo dal

punto di vista didattico e della scuola. Parlo soprattutto delle Conferenze dedicate alla Scuola e di quelle riservate al tema dell'Università. Nella seconda il discorso è discretamente omogeneo, nel senso che sono intervenuti per lo più orizzontali tra studenti dei vari atenei d'Italia, con scambi di opinioni, informazioni, consigli relativi alle scelte dei corsi. C'è anche qualche docente universitario che ogni tanto fa capolino, anche se si tratta di poche eccezioni. Nell'area Scuola, invece, il panorama dei messaggi è un po' più variegato e, al momento, discretamente ampio, quasi "general purpose".

D. Per il BTB, quali sono attualmente i progetti in cantiere da realizzare a breve e medio termine?

R. Il BTB attualmente è impegnato in varie attività, strettamente collegate ai diversi progetti dell'ITD nel quale si trova a operare. In breve, possiamo menzionare:

1) KIDSLINK. Progetto nato dall'idea di Odd de Presno, un docente

finlandese che ha esteso l'idea del "pen-pal", l'amico di penna, alla telematica e soprattutto alla possibilità di utilizzare le reti universitarie per far dialogare e incontrare telematicamente i giovani di domani, per iniziare a conoscersi in quello che domani sarà un mondo ancora più dilatato e meno preoccupato delle frontiere e delle distanze. L'attività è partita intorno a pochi elementi fissi: i ragazzini (rigorosamente under 15) dovevano presentarsi brevemente rispondendo a quattro domande (chi sono io, cosa vorrei fare da grande, come vorrei che fosse il mondo domani, cosa posso fare io adesso per realizzare questo mio ideale) e quindi trovare un interlocutore con il quale avviare un dialogo. Sono subito nate due conferenze: una per le presentazioni dei ragazzi (KidLINK) e una per lo scambio vero e proprio (KidCAFE), subito seguite da altre di coordinamento riservate ai docenti e ai responsabili. Attualmente, sono coinvolte centinaia di persone sparpagliate in tutto il mondo: sono già attivi collegamenti con tutti i Paesi occidentali, la Cina, la Thailandia, il Giappone, l'ex-Urss, il Kenya e quasi quotidianamente la rete si espande. Il traffico di questa sperimentazione è decisamente alto e si aggira sui 30-40 messaggi al giorno. Al momento, il collegamento è già attivo ma ancora poco noto e le realtà scolastiche coinvolte sono solo alcune scuole di Bologna. Il BTB, insieme al nodo ARCI di Bologna che tecnicamente pensa a smistare tutto il materiale, è uno dei punti di riferimento per l'Italia di questa iniziativa.

2) Collegamenti telematici tra le scuole di Genova. Nella realtà scolastica di Genova l'ITD ha svolto un ruolo guida per quanto riguarda la sperimentazione dell'informatica nella scuola dell'obbligo. Numerosi sono stati i laboratori-computer avviati e cresciuti con la collaborazione dell'Istituto, che si propone di continuare quest'attività proprio nell'ambito della telematica. È allo studio un progetto di collegamento dei diversi laboratori e un piano didattico che preveda lo scambio tra le classi su determinati argomenti.

3) Impegno con il Piano Nazionale Informatica per le materie Umanistiche (PN12). Nell'ambito del PN12 alcune persone che hanno lavorato al BTB dovrebbero essere nel gruppo dei formatori dei docenti per ciò che

riguarda la telematica applicata alla didattica, sia come contenuto che come strumento. In questa situazione, l'esperienza maturata negli anni di vita del BTB verrà utilizzata per implementare uno strumento che serva esclusivamente alla formazione a distanza dei docenti.

D. Possiamo dire agli insegnanti interessati (speriamo moltissimi) e a studenti altrettanto interessati (magari per raccogliere materiale per la loro tesi di Laurea) che possono contare sul BTB quale sicuro punto di riferimento per informazioni, scambio d'idee e opinioni sulle molteplici tematiche inerenti alla scuola e all'insegnamento?

R. A proposito di tesi di laurea, l'idea di mettere a disposizione qualche titolo di lavoro realizzato da studenti italiani sulle tematiche d'interesse del BTB era stata una delle proposte iniziali che, logicamente, ha incontrato l'ovvio ostacolo della reperibilità di questo materiale. Sarebbe interessante che studenti interessati a queste sperimentazioni e che hanno fatto la loro tesi su questi argomenti ci inviassero il loro lavoro registrato su file in modo da poterlo rendere disponibile. Ma vista la novità di queste sperimentazioni, è più facile che una tesi di laurea diventi una miniera per articoli e pubblicazioni varie piuttosto che un documento liberamente prelevabile da BBS. Al momento, abbiamo solo un paio di questi lavori, ma se gli interessati ci volessero mandare anche solo un abstract con i riferimenti per contattare l'autore... Per quanto riguarda invece la bibliografia e la documentazione varia sul BTB, si può sicuramente trovare della documentazione interessante e testi vari. In pratica, quando si riesce a trovare un documento di un certo rilievo o si trova il tempo per acquisirlo con lo scanner e passare all'OCR qualche articolo interessante, è sicuro che finirà nell'elenco file del BTB. Tra l'altro, c'è un discreto scambio con altri BBS che hanno materiale simile e il tam-tam telematico è molto efficace. Per non parlare delle persone che stanno svolgendo lavori di laurea proprio sullo sviluppo del fenomeno BBS!

Per chi volesse contattare l'ITD e il BTB, riportiamo l'indirizzo e l'identificativo di nodo: Istituto Tecnologie Didattiche, Via all'Opera Pia 11, 16145 Genova (BTB 2:332/208).

(a cura di Stefano Franzato) ■



PassoPasso

*L'Editor Passo Uno
per
Amiga*

- controlla VTR da editing di qualsiasi formato con porta RS422
- pilota, via Arexx, qualsiasi scheda grafica per Amiga
- controlla, via Arexx, programmi di elaborazione di immagine
- può usare l'uscita video di Amiga
- legge e genera direttamente il Time-Code
- costruisce sofisticate sequenze di montaggio
- è prodotto in Italia

DeskTop Video srl
via Quarnero 14 - 20145 MILANO

COMPUTER CENTER

VENDITA PER CORRISPONDENZA

PER INFORMAZIONI E/O ORDINAZIONI:

Via Forze Armate, 260 - 20152 Milano

Tel. 02/48016309 - 4890213

Fax 02/4890213

HARDITAL

SHOW ROOM

VENDITA DIRETTA

Via G. Cantoni, 12

20144 Milano

Tel. 02/4983457 - 4983462

Fax 02/4983462 - Hot Line 0337/345899

**TUTTI I PREZZI
SONO
IVA COMPRESA**

NOVITÀ DEL MESE

Amiga 4000-30 4 MB- 80 MB garanzia Commodore Italia	L. 2.290.000
come sopra ma con coprocessore 68882 25 MHz	L. 2.430.000
Scheda di memoria a 32 bit esp. a 8 MB con zoccolo per coprocessore matematico per A1200 4 MB	L. 429.000
Completa di coprocessore 68882 25 MHz	L. 570.000

POWER CHANGER

Scheda acceleratrice per Amiga 4000. Si inserisce sullo zoccolo CPU.	
Power Changer 040-28 con CPU 68040 a 28.5 MHz	L. 890.000
Power Changer 040-33 con CPU 68040 a 33 MHz	L. 1.390.000
Power Changer 030 con CPU 68030 a 40 MHz	L. 490.000
Power Changer 030-82 come sopra più coprocessore 68882 a 40 MHz	L. 690.000
Modulo Simm 32 bit 4 MB per Amiga 4000	L. 319.000

HD SCSI CONTROLLER PER A2000

Synthesys 2000 Hardital SCSI-2, esp. 8 MB (0 Ram)	L. 170.000
ADSCSI 2000 ICD SCSI-2	L. 140.000

HD SCSI CONTROLLER PER A500

Synthesys 500 Hardital Esterno esp. 8 MB controller SCSI 2	L. 199.000
Per ogni MB aggiungere	L. 100.000

HARD DISK SCSI

ELS 43 MB Quantum	L. 390.000
ELS 85 MB Quantum	L. 520.000
ELS 127 MB Quantum	L. 650.000
LPS 240 MB Quantum	L. 990.000
LPS 425 MB Quantum	L. 1.640.000

HARD DISK ATBUS PER A600-A1200

43 MB 2,5"	L. 360.000
85 MB 2,5"	L. 540.000
120 MB 2,5"	L. 740.000
cavetto 44 poli.	L. 39.000

I COMPUTER AMIGA

A600 garanzia Commodore Italiana	L. 489.000
A600 HD40 come sopra ma con HD 30 MB	L. 820.000
A1200 Commodore Italiana	L. 740.000
A2000 garanzia Commodore Italiana	L. 949.000
A3000-25-52 25 MHz con HD 52 MB	L. 1.990.000
A4000-040 Commodore Italiana	
6 MB HD 120 MB.	L. 3.849.000

I DISCHETTI

FDD 3,5 dischi Bulk 3,5 880K (l'uno)	L. 720
FDD 3,5 HD Bulk 3,5 1.44 MB (l'uno)	L. 1.190

SCHEDE AUDIO-VIDEO

Merlin-XPert potentissima scheda grafica a 24 Bit per Amiga 2000-3000 e 4000. Risoluzione fino a 2048 x 2048. Si inserisce nello slot Zorro II o III a 32 Bit. Pip a 24 Bit in tempo reale. Compatibilità con 3.0 e 68040 ed emulazione Chip Set AA. Emulazione via software anti flicker. Ingressi PAL/NTSC - FBAS - Y/C - SVHS - RGB - uscite RGB per qualsiasi tipo di monitor registratore video. Incluso software TV Paint 2.0	
Merlin 1 MB	L. 598.000
2 MB	L. 689.000
4 MB	L. 848.000
Modulo Genlock 24 Bit.	L. 598.000
Modulo digitalizzatore 24 Bit.	L. 598.000
Entrambi i moduli	L. 798.000
A2320 Commodore Deinterlace Card per A-2000	L. 370.000
FLICKER FIX Flicker Fixer per A-2000	L. 310.000
FLICKER FIX 500 Flicker Fixer per A-500	L. 310.000
A 2320 Commodore genlock card per A-2000 interna	L. 279.000
A 520 Modulatore Tv est. per 500/2000	L. 41.000

SCHEDE ACCELERATRICI

BANG 2081 68020/16 MHz	
------------------------	--

per A500, 500+, 2000	L. 199.000
BIG BANG 25/1 68030 32 bit esp. a 8 MB interna per A500, 500+, 2000	L. 340.000
SUPER BIG BANG 25 68030 con controller SCSI-2 esp. a 8 MB per A2000	L. 490.000
RAM a 32 bit per BIG e SUPER BIG BANG ogni MB	L. 100.000
OVER THE TOP 68040 esp. a 32 MB Ram per A2000 (0 Ram)	L. 990.000
OVER THE TOP 68040 esp. a 32 MB Ram esterna per A500	L. 1.090.000
OTTRAM Ram a 32 bit per OVER THE TOP ogni 4 MB Ram	L. 360.000
Modulo controller SCSI 2 per OVER THE TOP	L. 199.000

Dominator 882-scheda per A1200 con coprocessore e orologio

con 68881 a 25 MHz	L. 199.000
con 68882 a 25 MHz	L. 280.000
con 68882 a 33 MHz	L. 340.000
con 68882 a 50 MHz	L. 480.000

Coprocessori per schede acceleratrici BANG, BIG BANG, SUPER BIG BANG e DOMINATOR

68881 25 MHz	L. 99.000
68882 25 MHz	L. 180.000
68882 50 MHz	L. 340.000

Coprocessori per UXL 30 A1200 e A4000 -30	
68882 25 MHz	L. 149.000

PROCESSORI

CPU 68000 A 16 MHz	L. 29.000
CPU 68010	L. 24.000
68020 16 MHz	L. 140.000
68030 25 MHz	L. 230.000
68030 50 MHz	L. 390.000
68040 25 MHz	L. 800.000

DRIVE

ADRIIVE drive esterno da 880K per tutti i modelli Amiga	L. 105.000
AD2000 drive interno da 3,5" 880K per A2000	L. 99.000
A570 Comm. CD ROM per A500 e 500+	L. 570.000
SUPERDRIVE drive esterno per tutti i modelli Amiga selezionabile da 880K a 1,640 kb.	L. 169.000

ESPANSIONI PER A2000

Synthesis Hardital 2 MB	L. 340.000
4 MB	
8 MB	

ESPANSIONI PER A500

INSIDER 0,5 (0,5 MB interna per A500)	L. 59.000
INSIDER 0,5C (come sopra ma con clock)	L. 74.000
INSIDER 1 (1 MB interna per A500+)	L. 89.000
INSIDER 2 (2 MB interna con clock per A500)	L. 199.000
INSIDER 4-1 (4 MB interna con clock per A500 con 1 MB)	L. 169.000
INSIDER 4-2 (come sopra ma con 2 MB)	L. 240.000

INSIDER 4-4 (come sopra ma con 4 MB)	L. 360.000
Espansioni per A600 e A1200 H601	
1MB int + A600	L. 89.000
Xpander 2MB per porta PCMCIA	L. 320.000
Come sopra ma con 4MB	L. 420.000

ESPANSIONI CHIP RAM PER A500 E A2000

Mega Agnus Hardital 2MB di chip Ram	L. 349.000
-------------------------------------	------------

ESPANSIONI PER A3000

Ram Zip 1 MBx4-2MB	L. 200.000
4 MB	L. 380.000
8 MB	L. 640.000

I MONITOR

1084S Commodore colori 14" stereo per tutti i computer Amiga	L. 420.000
1960 Commodore multisync colori 14" per tutti i modelli Amiga	L. 780.000
HARSVGA monitor 14" SVGA 1280 x 1024 per schede flicker	L. 550.000

LE STAMPANTI

MPS 1230 (80 colonne b/n)	L. 310.000
MPS 1270 (80 colonne inkjet)	L. 280.000
MPS 1550 (80 colonne colori)	L. 410.000
CANON BJ10 (80 colonne bubble jet)	L. 639.000
CANON LPB4 (laser 4PPM)	L. 1.550.000

PC IBM COMPATIBILI

PC 386-SX33 Mhz Computer con CPU 386-33 (LM 33 Mhz) 1 MB Ram - floppy drive 1,44 MB scheda video VGA 1 MB (1.024x768) - controller IDE per 2HD e 2FD - Uscite: 2 seriali - 1 parallela - 1 game - Case Desktop o Minitor con display - Tastiera estesa 102 tasti	L. 649.000
PC 386-40 Computer con CPU 386DX-40 (LM 67 Mhz) Cache 64 KB - 4 MB Ram - Floppy Drives da 1,44 MB - Scheda Video VGA 1 MB (1.024x768) - Controller IDE per 2 HD 2FD - Uscite 2 seriali - 1 parallela - 1 game - Case Desktop o Minitor con Display - Tastiera estesa 102 tasti	L. 1.075.000
PC 486DLC-33 come sopra ma basato su CPU 486DLC33-33 (LM 117 Mhz)	L. 1.099.000
PC 486DX-33 local bus come sopra ma basato su CPU 486DX-33 (LM 151 Mhz) - cache 128 KB	L. 1.590.000
PC 486DX-50 local bus come sopra ma basato su CPU 486DX-50 (LM 230 Mhz) - cache 128 KB	L. 1.990.000
PC 486DX2-66 local bus come sopra ma basato su CPU 486DX2-66 - cache 64 KB	L. 2.290.000
NOTEBOOK 386SX-25 Notebook con CPU 386SX33 con display LCD retroilluminato VGA 640x480 - 4 MB Ram - 1 Drive da 1,44 MB - 1 Hard Disk da 60 MB - Batterie - Borsa trasporto - Uscite: 2 Seriali - 1 Parallela - Monitor esterno - Tastiera esterna - Dim. 28x22x4 - Peso Kg. 2,4	L. 1.990.000

VARIAZIONI ALLE SOPRAELENATE CONFIGURAZIONI

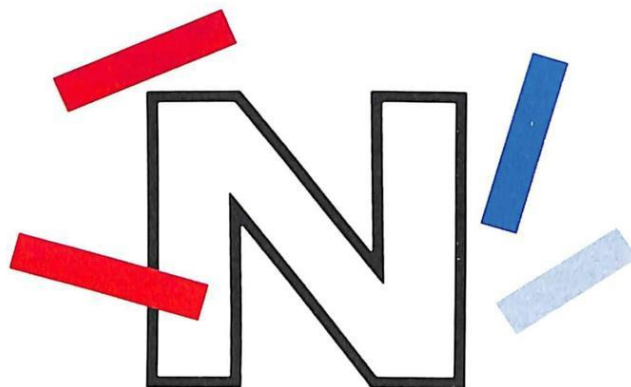
Per ogni MB di RAM aggiuntiva	L. 89.000
Scheda VGA 1 MB (1024x768) - 16 M col. e acceleratore grafico +	L. 75.000
Come sopra ma local bus +	L. 99.000

PERIFERICHE

Hard Disk 120 MB - 3,5" IDE AT	L. 410.000
Hard Disk 130 MB - 3,5" IDE AT	L. 460.000
Hard Disk 210 MB - 3,5" IDE AT	L. 620.000
Monitor 14" - B/N - VGA	L. 190.000
Monitor Samsung 14" colori - VGA (1024x768x0.28 pitch)	L. 550.000
Monitor 20" colori - VGA (1280x1024x0.28)	L. 1.990.000
Scheda Sound Blaster Pro	L. 289.000
Mouse a 3 tasti	L. 29.000

COMPUTER NEWS

NOVITÀ HARD E SOFTWARE DALL'ITALIA E DAL MONDO



ITALIA

CITIZEN E IL DRIVER AMIGA

La Citizen ha introdotto sul mercato un nuovo driver software destinato agli utenti che impiegano stampanti Citizen con l'Amiga. Il driver è basato sul pacchetto Irseesoft Turbo Print Professional, costa 30 mila lire + Iva e consente una migliore qualità di stampa, soprattutto a colori. Può essere usato con qualsiasi stampante Citizen, dalla 120D+ ad aghi alla nuova Project a getto d'inchiostro. Per informazioni si possono contattare i due distributori dei prodotti Citizen per l'Italia: Telcom (Tel. 02/48302640) ed Eletec (Tel. 02/48200686).

NUOVI TITOLI PER CDTV

La Commodore (Tel. 02/661231) ha segnalato la disponibilità dei seguenti nuovi titoli per CDTV: *Mind Run* (test); *Cinderella, Original Fairy Tale* e *Language Tv English* (programmi didattici per la lingua inglese); *Nasa* (archivio di dati e immagini); *North Polar Expedition* (gioco di ruolo + programmi educativi); *Fantastic Voyage* (gioco tratto dal film omonimo); *Prehistorik* (gioco arcade); *Battle Chess* (i celebri scacchi); *Trivial Pursuit* (la versione al computer del celebre gioco da tavolo); *Ultimate Basketball* (gioco); *Sherlock Holmes* (sequenze video in movimento, immagini e suoni in un gioco da detective); *Team Yankee* (siete al

comando di quattro plotoni di carri armati); *Tie Break* (simulazione del tennis); *The Curse of Ra* (rompicapo strategico); *Turrican I & II* (noto spara e fuggi); *Logical* (spremi meningi con mattoncini colorati da collocare).

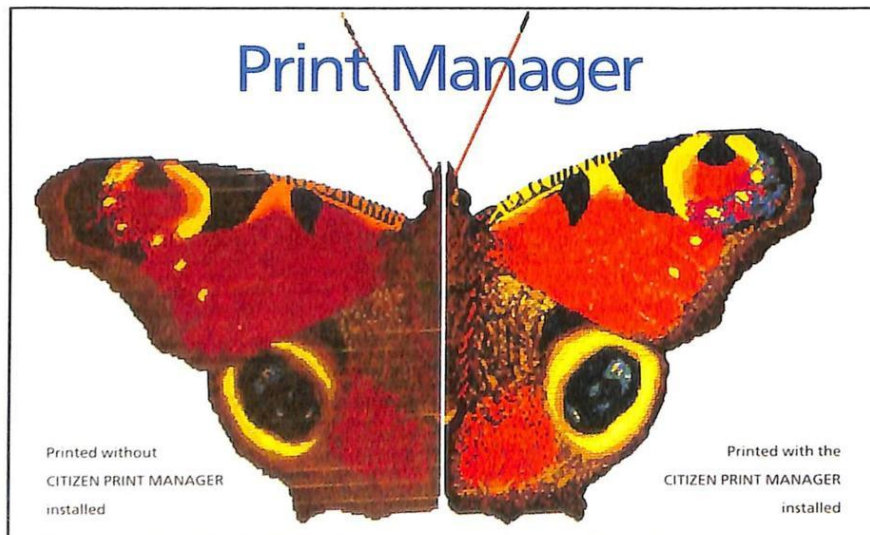
AMIGA HALL OF FAME

Alla Conferenza dei programmatori di Orlando (Florida, USA), sono stati premiati i migliori programmi del '92: *SAS/C 6.0* del SAS Institute, *Deluxe Paint IV* della Electronic Arts, *The Miracle piano teaching system* della Software Toolworks e infine come gioco *Pinball Fantasies* della 21st Century Entertainment.

ESTERO

PAGESTREAM 3.0

La Soft-Logik (Tel. 001/314/8948608) ha presentato la nuova versione 3.0 di *PageStream*, programma di desktop publishing. Ecco alcune delle novità: tag override, tag ripple, object tag; dot leaders (filled tabs); allineamenti a destra, sinistra e al centro di qualsiasi carattere; importazione di file Word Perfect, IFF ILBM, Adobe Illustrator 3.0 EPS; compatibilità HotLinks, ARexx, AGA, ASL e AmigaGuide. *PageStream 3.0* sarà disponibile quest'estate al prezzo di 299.95 dollari.



PAGINE GIALLE

**Dove acquistare il vostro hardware e software,
dove far riparare il vostro computer**

LOMBARDIA

Computer Lab - Via Cadore, 6 - 20135 Milano - ☎ (02) 54.64.436 - Fax (031) 54.65.036. Centro assistenza autorizzato Commodore ed Epson. Riparazioni in e fuori garanzia. Installazioni ed aggiornamenti con prodotti originali delle migliori marche. Banca dati: (02) 55.01.91.50 - +

SOFTMAIL

SoftMail è dal 1984 il n. 1 della vendita per corrispondenza di software ricreativo e accessori per i computer più diffusi. Richieda oggi il catalogo gratuito: Le verrà spedito immediatamente!

SoftMail - Via Napoleona, 16 - 22100 Como - ☎ (031) 300.174 Fax (031) 300.214 - ☎ ☎

Punto di vendita diretta al pubblico aperto lu-ve 9/13 e 14/18

VENETO

Carpanese Elettronica - St. Sette

Martiri, 101 - 35143 Padova - ☎ (049) 62.41.60 - +

LIGURIA

C.S. COMPUTER SERVICE

Assistenza tecnica specializzata: Commodore Amiga 500, 2000, 3000 Stampanti di tutte le marche Monitor - PC compatibili

Piazza P. da Novi, 13-15R
16129 Genova
☎ (010) 55.31.744 - ☎ ☎

EMILIA ROMAGNA

Computer House - Viale Tripoli, 193/D - 47037 Rimini - ☎ (0541) 39.13.62 - ☎ ☎ ☎

Ravezzi Angelo - Via Avogadro, 6/10 - 47037 Ravazzura di Rimini (FO) - ☎ (0541) 37.36.86 - +

TOSCANA

Elettronica Centostelle srl - Via Cen-

LEGENDA:

- ☎ Vendita diretta
- ☎ Vendita per corrispondenza
- + Centri di assistenza per le riparazioni

tostelle, 5 A/B - 50137 Firenze - ☎ (055) 61.02.51 - 60.81.07 - ☎ ☎

Electronic Dreams - Via Dante, 77 - 56025 Pontedera - ☎ (0587) 52.063. Accessori per Amiga & compatibili. Riparazioni, consolle per videogames, giochi originali. Vendita per corrispondenza. GVP Point - ☎ ☎ ☎

MARCHE

MGA Computer - Corso Mazzini, 23 - 63039 S. Benedetto del Tronto (AP) - ☎ (0735) 58.34.54 - +

ABRUZZO

MGA Computer - Via Trilussa, 24 - 65122 Pescara - ☎ (085) 42.15.599 - +

ATTENZIONE!

I rivenditori e i riparatori interessati a essere inseriti in questa rubrica possono richiedere il relativo modulo di adesione alla rubrica PAGINE GIALLE, telefonando allo 02/794181 - 76022612, inviando un fax di richiesta allo 02/784021 oppure scrivendo a:

**Commodore Gazette
Rubrica Pagine Gialle
Via Monte Napoleone, 9
20121 Milano**

PUGLIA

Williams Computer Center - Viale Unità d'Italia, 79 - 70125 Bari - ☎ (080) 53.63.579 - ☎ ☎ ☎

SICILIA

Azeta s.r.l. - Via Canfora, 140/142 - 95128 Catania - ☎ (095) 72.77.620 - ☎ ☎ ☎

New Systems - Corso Umberto I, 321 - 90048 San Giuseppe Jato (PA) - ☎ (091) 85.78.344. Vendita per corrispondenza in tutta Italia, spedizioni entro 48 ore. Computers Amiga e PC, floppy disk, accessori, consolle. Esempi: Amiga 1200 garanzia italiana L. 750.000, Super Nes L. 370.000. I prezzi sono IVA compresa - ☎ ☎ ☎

Db-Line Srl

La tecnologia nel commercio

bsc

Distributore ufficiale per l'Italia dei prodotti **bsc** per Amiga

Riservato ai rivenditori

Db-Line mette a disposizione dei rivenditori un'efficiente staff di vendita per soddisfare ogni necessità nel campo hardware di Amiga e dei CD-ROM. La linea di prodotti **bsc** comprende: controllers SCSI-2 ed At-Bus per Amiga 500, 2000, 3000 e 4000, schede grafiche da 4096 e 16 milioni di colori, digitalizzatori a 24 bit, schede multiseriali velocissime, espansioni di memoria per ogni modello Amiga (anche il 600), schede di rete per Novell e software per gestire scanners, memoria virtuale ed altro ancora!

VOXonFAX: Dal telefono del vostro fax chiamate lo 0332/767360. Vi risponderà VOXonFAX che Vi fornirà in automatico 24 ore al giorno informazioni sui nostri prodotti e disponibilità. Richiedete il Vostro codice di accesso chiamando lo 0332/767270 o inviando un fax con i dati della vostra ditta allo 0332/767244.

Si cercano agenti per le zone libere

Posta: V.le Rimembranze, 26/C - 21024 Biandronno (VA)
Tel.: 0332/767270 r.a. BBS: SkyLink 0332/706469-739
FAX: 0332/767244 Db-Line 0332/767277-329

VOXonFAX: 0332/767360

Computer Service
di Antonio Piscopo

Corso A.Lucci, 137 - 80142 Napoli - Tel/Fax 081/5536257

Unica sede



Rivenditore autorizzato per Napoli e Campania dei prodotti GVP

Electronic Design LOGICA OpalVision

La nostra azienda è specializzata nella fornitura di sistemi per la titolazione video, l'animazione 2D e 3D, ed il DVE. Presso di noi troverete la migliore produzione mondiale di software, accessori e periferiche per tutta la linea Commodore AMIGA.

Computer Service è anche un qualificato centro di assistenza tecnica su tutte le macchine COMMODORE, AMIGA, IBM compatibili.

Preventivi gratuiti e riconsegna rapidissima.

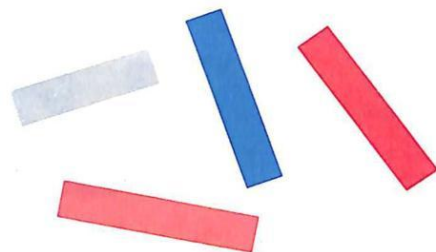
Effettuiamo pagamenti dilazionabili da 6 a 48 mesi.

NOVITA' NOVITA' NOVITA' NOVITA' NOVITA'

dpaintBOX®

In esclusiva per l'Italia la nuovissima tavoletta grafica per Amiga che si interfaccia direttamente a DeluxePaint. Software e manuale in italiano.

CLASSIFIED



Software

Vendo Amos Basic versione 1.2 + Monkey Island a Lire 60.000 + spese postali. Tel. 0565/225479 - Giuseppe Galante (ora di cena).

Computer Club Napoli. Distribuzione software di pubblico dominio e shareware, per Amiga & MS-DOS. Vasta biblioteca di software e manuali. Per informazioni telefonare a Massimiliano (ore pomeridiane) - Tel. 081/7284810.

Lotto 6.00 programma originale per PC-Windows e Amiga gestisce ritardi frequenze, decine, cadenze, figure, etc. con archivio estrazioni dal 1939 ad oggi e ricerche, previsioni, progressioni. Solo Lire 35.000. Massimo Chiales - Via Roma,

VENDITA ECCEZIONALE

Hardware e software originale per Amiga: digitalizzatore audio Technosound, Sim heart, Shuttle, KCS 3.5, Professional Page 3.0, Take-2, B&P Pro 1.0e telefonare

Scheda Impact Vision 24 GVP + VIU-S + adattatore per A2000 + VIU-CT (splitter professionale), eccezionale scheda a 24 bit professionale per A2000/3000 con de-interlaser, digitalizzatore, PIP e genlock completa di software Scala, Caligari e Macropaint. Nuova, mai usata, con garanzia da spedire Telefonare

Commodore 64, disk drive 1541, stampante Letter Quality, stampante Rietman C+, circa 5 mila programmi per C-64/128 di cui molti con confezioni originali, Polaroid Palette, Amiga 1000 NTSC Ai migliori offerenti

Complete Colour Solution della Rombo per PC MS-DOS (digitalizzatore, framegrabber, splitter, software...), nuovo, mai usato L. 490.000

Video Director della Gold Disk per Amiga interfaccia + software per trasformare il computer in una centralina per il montaggio di videocassette, come nuovo L. 300.000

Noise gate Vestafire nuovo L. 120.000

Scheda controller HD + RAM Impact II della GVP Al miglior offerente

Genlock G-Lock della GVP nuovo, ancora sigillato nell'imballo originale Al miglior offerente

Lettore di CD Sony multipiatto per 5 dischi, come nuovo con telecomando L. 300.000

Registratore professionale a bobine Fo-ster E8 a 8 piste, come nuovo (eventualmente posso fornire anche il relativo mixer) L. 4.500.000

DAT Aiwa da tavolo XD-S1100 ancora in garanzia L. 750.000

Processore stereo surround Marantz SP-35 ancora in garanzia L. 250.000

Software C-64 GEOS originale con scatola e manuale: Desktop Plus, Geodex, Geopublish, Geocalc, Geospell, Geos 128 e Geos 128 2.0, e inoltre SuperForth ai migliori offerenti

Telefonare dopo le ore 20.30 a Massimiliano: 02/86460434.

123 - 14019 Villanova (AT) - Tel. 0141/948015.

Hey Amighi! Problemi di virus? Niente paura, chiamate "Safe Hek International". Abbiamo i più recenti antivirus shareware per Amiga con manuali anche in italiano. Massimo Gais, Via Vittorio Veneto, 21 - 80029 S. Antimo (NA) - Tel. 081/5052256 (ore serali).

Cercasi Devpac Assembler per Amiga (usato) compreso di manuali a buon prezzo. Telefonare ore pasti allo: 0565/225479 - Giuseppe.

Scambio, vendo programmi e dati di astronomia per PC-Windows, Amiga, Mac e altri personal e sistemi operativi (Windows, Unix). Dispongo di vasto archivio (20 MB) di programmi e database, PD e Shareware. Scrivete a: Alessandro Freda - Via Maona, 3/10 - 16155 Genova-Pegli.

Se sei stanco del solito software pirata, prova il Pubblico Dominio: prova i famosi Fish Disk per Amiga. Oltre 800 dischi sempre in aumento con il meglio di utilities, giochi e programmi di tutti i tipi e gusti. Lire 2.500 a disco con sconti quantità. Possibili abbonamenti. Scrivi a: Michele Masiero - Via Bracciano, 26 - 35030 Selvazzano (PD).

Cercasi programmi AREXX per il controllo (in "Scala MM" via AREXX) delle varie funzioni di controllo del Glock GVP e inoltre i programmi "Scape Maker", "Video Studio" e "Terra Form". Contattare: Luigi Onofrio - Tel. 0823/444530.

Hardware

Vendo Amiga 600HD, 20 MB + manuali in italiano n. 3 + 4 dischi di sistema 2.05 + mouse + copri computer + giochi e programmi. Garanzia Commodore Italiana, un mese di vita. Prezzo: Lire 800.000 trattabili. Daniele - Tel. 0761/443278 (sabato pomeriggio).

Vendo Amiga 500 Plus in garanzia + monitor colori 10845 + drive A1011 + mouse e manuali.

CLASSIFIED È UNA RUBRICA DI PICCOLA PUBBLICITÀ GRATUITA TRA PRIVATI. PER INSERIRE IL VOSTRO ANNUNCIO DOVETE COMPILARE E SPEDIRE IL MODULO PUBBLICATO A PAGINA 95-96.

Il modulo va spedito in originale, non si accettano fotocopie.

Gli annunci sono soggetti all'approvazione dell'Editore.

La Direzione del periodico non si assume responsabilità in caso di reclami di qualunque natura da parte degli inserzionisti e/o dei lettori. Nessuna responsabilità è altresì accettata per errori e/o omissioni di qualsiasi tipo. La responsabilità del testo e del contenuto dell'annuncio è dell'inserzionista.

Regalo 100 dischi - software di grafica, video scrittura, giochi + 2 joystick. Tutto a prezzi ridicoli. Ondino - Tel. 0735/656445 o 085/8998121.

Vendesi: scheda acceleratrice Commodore A2630 (68030 + 68882 a 25 MHz) con 4 MB Ram a 32 Bit a Lire 800.000. Scheda interfaccia GVP SCSI + 2 MB Ram + HD Quantum 80 MB a Lire 600.000. Giacomo - Tel. 010/3621473.

Vendo monitor Commodore 10845 stereofonico a Lire 300.000. Telefonare o scrivere: Enzo Pizzo - Ctr. M.C. Bufalata, 301 - 91025 Marsala (TP) - Tel. 0923/967058

Cerco monitor colori per Commodore 64 a poco prezzo. Viviana Bompreszi - Via Diaz, 53 - c/o U.I.C. - 61100 Pesaro.

Janus Scheda A2286, kit drive esterno 3 1/2 da 1,44 MB + drive 5 1/4 da 1,2 MB, eventuale VGA, MS-DOS 5, PCTools v. 6. Vendo. Tel. 0574/790711 - Mario (ore pasti).

Vendo Amiga 3000 16 MHz, hard disk 52 MB, 6 MB di Ram, monitor VGA Philips, digitalizzatore audio, 2 drives, confezioni e manuali originali, sistema operativo 2.1 in italiano, numerosi programmi e giochi a Lire 2.500.000. Tel. 02/9340337 - Dario.

Cerco Kickstart/Workbench v. 1.3 per Amiga 2000-A (1ª versione commercializzata), che permetta l'uso dei programmi in commercio odierno, possibilmente via software. In caso contrario di hardware. Annuncio sempre valido. Tel. 0861/712133 - Angelo (c'è la segreteria).

Vendo Amiga 2000B 1 MB Ram 2 FDD + monitor 1084 + interfaccia televideo + Tuner TV Philips + casse stereofoniche + 2 joystick + un numero imprecisato di dischetti con giochi e programmi vari di musica, utilità, grafica, WP, ecc... Prezzo da concordare. Antonio Di Palma - Via Petrosini, 10 - 84014 Nocera Inferiore (SA) - Tel. 081/5171391 (ore pasti e serali).

Vendo Amiga 2000B con Kickstart 1.3, 1 MB di Ram, scheda Janus A2088, monitor 1084S stereo, secondo drive esterno A2010, joystick e software. Fate le vostre offerte telefonando ore pasti allo: 0521/582786 - Giacomo.

Vendo per Amiga 500/500+ cartuccia Action Replay II completa di manuale ed anche copritastiera. Vendo inoltre collezione completa di The Games Machine dal n. 10 in poi e di K dal n. 18 in poi (anche solo parti della collezione). Davide Grangia - Via Monteverde, 20 - 15033 Casale Monferrato (AL) - Tel. 0142/452882.

Vendo varie: Amiga 2000 v. 1.3 con 1 MB di Ram e 2 drives da 3 1/2; Genlock interno; Amiga Action Replay 3 per A2000. Anche separatamente. Telefonare dalle 19.30 alle 20.30 allo: 0175/85924 - Fabio.

Vendo Amiga 2000, controller 2091 con 2 Mega, Atonce Plus, secondo drive, monitor 1084S. Anche separatamente. Ore serali. Tel. 0961/701825 - Salvatore. Vero affare. ECS 2.0 + 1.3.

Cerco Hard disk ed espansione di memoria per A1000. Prego contattare di nuovo lo 051/260075 causa messaggio troncato in segreteria telefonica (dopo le ore 20.00. Verrete richiamati se non troverete nessuno). Cristiano Nanni.

Vendo 8 MB di Ram per Amiga 3000, Static Column 80ns a sole Lire 1.000.000. Digitalizzatore Digiview Gold 4.0 per Amiga 500/1000/2000 a Lire 150.000. Tel. 0365/598757.

Vendo Amiga 2000 1 MB Chip Ram con espansione GVP 2 MB, floppy drive esterno Roctec in perfetto stato a Lire 600.000. Carlo Cappa - Via Cavalli, 4 - 25078 Vestone (BS) - Tel. 0365/820151 (ore serali).

Vendo scheda di acquisizione video per sistemi MAC II con slot Nubus "Computer Eyes Pro" 16 milioni di colori, 24 bit. Mai utilizzata e ancora nell'imballo originale a Lire 500.000. Telefonare a: Marco Antonio Campolo - Tel. 0965/713007 (ore serali).

Vendo PC 386 DX, 4 MB Ram, SVGA 1 MB, 4 MB Ram, 100 MB HD, Sound Blaster, 2 drive 5" e 3", monitor 15" 1024 non interlacciato + DOS, Windows e altri programmi originali a Lire 2.500.000 trattabili. Tel. 030/9130450 - Carlo.

Vendo A2000 Kickstart 1.2/2.04 con switch esterno, Workbench e manualistica originale a Lire 500.000. Monitor Philips CM8833 colori stereo a Lire 200.000. Christian Biasini - Tel. 02/6108424.

Vendo scheda grafica 24 Bit Colourburst + digitalizzatore Videon III. Tutto con Manuale software e imballi a Lire 600.000 trattabili. Francesco - Tel. 010/566167 (serali).

Vendo Commodore Amiga 600HD con hard disk da 30 MB, ancora in garanzia allo strepitoso prezzo di Lire 699.000. Regalo alcuni giochi originali più Imagine, Real 3D, Scala Multimedia, Bars & Pipes Pro., Caligari 3D, ecc. Antonio - Tel.

0883/533405.

Vendo A500 v.1.3 + manuali + joystick + software a Lire 250.000. HD controller Rochard + HD Fujitsu 45 MB formattato (4 mesi di vita) a Lire 500.000. Tel. 041/4966723 - Paolo (ore serali).

Vendo Amiga 500 in ottimalissime condizioni, espanso 1 MB, completo di manuali, mouse, 2 joystick, Paddle, penna ottica, tappetino + monitor Commodore 1084 e moltissimi programmi al miglior offerente. Alex Serafini - Via Emilia, 2 - 64026 Roseto degli Abruzzi (TE) - Tel. 085/8995212.

Vendo Amiga 500 plus ECS 2 mega chip Ram con hard disk Commodore A590, 20 MB con 2 mega fast Ram installati. Sistema operativo 3.0 con switch per passare al 2.0, il tutto a Lire 1.000.000, non trattabili. Garanzie ancora valide. Tel. 0733/506376 - Valerio (Macerata).

Amiga 2000HD Advanced, 68010, ECS, 3/8 MB, controller GVP, HD Quantum 52 MB, doppio drive, monitor, tavoletta grafica prof., scanner, imballi e manuali originali, 200 dischi, libri e altro a Lire 2.400.000. Tel. 02/4470267 - Alex.

Vendo Amiga 1200 con HD 80 mega, due mesi di vita, zona Milano. Telefonare ore pasti allo: 0371/425791 - Federico.

Vendo Amiga 2000 Hardital Super Big Bang 68030-68882 40 MHz, 8 MB Ram, 32 bit; Megachip 2000, 2 MB chip Ram, floppy esterno, espansione 8 MB fast, Minigen, Genlock Y/C Electronic Design, digitalizzatore Videon 3.0, scheda grafica Colorburst 24/48 bit, 2 hard disk installati da 50 e 120 MB. Tel. 0331/596358 -

Daniele.

Vendesi A2000B v. 6.2 con doppio drive + 1 MB chip Ram + 2 MB Fast Ram a 32 bit + scheda acceleratrice 68030 + 68882 a 25 MHz. Il tutto completo di manualistica e imballi originali a Lire 1.500.000. Tel. 0832/612941 - Marco.

Vendo CDTV: lettore CD + 3 CD (World Vista, Wrath of the Demon, Enciclopedia Grolier), tastiera, mouse trackball, drive esterno, digitalizzatore video VIDEON e stampante b/n Citizen 120D. Prezzo da concordare. Telefonare ore pasti: Tel. 085/4491375 - Luigi.

Vendo A2000B v. 2.0 ECS 2 drives interni a Lire 700.000 + scheda acceleratrice GVP G-Force 030/882 25 MHz, 1 MB, 32 bit con HD Quantum 120 MB, 7 ms. a Lire 1.350.000 trattabili. Vendo in blocco o separatamente. Tel. 091/934939 - Giuseppe (ore pasti).

Vendo scanner Alfa Scan per Amiga 400 DPI + favoloso software Migraph OCR + Touch UP ultime versioni. Tutto a Lire 300.000. Francesco Toscano - Via C. Correnti, 2 - 20025 Legnano (MI) - Tel. 0331/597110

Vendo Videon III + Genlock Rockgen + Scanner bianco e nero 400 DPI a Lire 500.000. Telefonare allo 089/253155 - Nicola (ore 11-13 e 15-20).

Amiga 500 2 MByte RAM (con mouse e modulatore TV) a Lire 350.000 + hard disk Supra Drive 52 MByte e 2 MByte RAM (S.O. 1.3 e 2.0) con 100 MByte di software a Lire 650.000. Stampante Star LC10 a Lire 200.000. Emulatore PC AT ATonce Plus (con DOS 5.0 e Windows 3.1) a Lire 250.000. Tel. 0322/249446 - Simone (ore serali).

SIAMO I PROFESSIONISTI PIÙ SERI ED AFFIDABILI

A1200 L.769.000

A4000/030-HD 80 L.2.489.000

A4000/040-HD120 L.3.995.000

1960 MONITOR x A4000 L. 839.000

GARANZIA COMMODORE ITALIANA - IVA COMPRESA

GVP GAMMA COMPLETA
GENLOCK - DIGITALIZZATORI
HARD DISK - ESPANSIONI

COLLAUDIAMO

OGNI COMPONENTE PRIMA DELLA PARTENZA

SPEDIZIONI ACCURATISSIME

CORRIERE ESPRESSO ASSICURATO

HI-FI CLUB

CONCESSIONARIO UFFICIALE

Commodore
Collegno - TORINO

C.so Francia 92/c Tel. 011/4110256 (r.a)

TecnoShop® by data office s.a.s.

Via Roma, 5/7 - 80040 S. Sebastiano al Vesuvio (NA)

Tel. 081/5743260 Pbx - Fax 081/5743260

Commodore 64 + 2 joystick + omaggio	249.000	CYCLONE NEW Ultima versione del potente duplicatore hardware con soft di gestione e verifica copia L. 150.000
Amiga 600 - 68000 - 1 Mbyte chip ram	499.000	
Amiga 600 - 68000 - 1 Mbyte chip ram Hard Disk 40 Mbyte	869.000	
Amiga 1200 - 68020 - set chip AGA - 2 Mbyte chip ram	749.000	
Amiga 3000 - 68030 - 2 Mbyte ram - Hard Disk 50 Mbyte	2.699.000	
Amiga 4000 - 68040 - 6 Mbyte ram - HD 213 Mb - Drive 1.76 Mb	3.990.000	
Commodore Mps 1230 - Stampante 9 aghi b/n	299.000	
Commodore Mps 1270 - Stampante getto d'inchiostro b/n	269.000	
Commodore Mps 1550C - Stampante 9 aghi colore	399.000	
CD Rom per Amiga 500	620.000	
Monitor Commodore 1084s a colori Stereo	399.000	

Stampante Hewlett Packard 500 C 300 dpi
Stampa a colori su carta normale mediante
getto termico d'inchiostro
L. 1.099.000

GVP Controller SCSI per 1200 con
4 Mb ram espandibile fino a 8 Mb
+ zoccolo per coprocessore
L. 1.299.000

GVP A1230 turbo
Interfaccia interna per Amiga 1200
con 68030 40 MHz 1Mb ram 32bit esp.
fino a 32 Mb con zoccolo per coprocessore
L. 1.199.000

HD per A1200 A600
alcuni esempi:
Conner 60 Mb L. 600.000
Seagate 120 Mb L. 800.000

Vasto assortimento
di software originale
per Amiga e Pc

**Stampante
Citizen
Swift 240C**

Dischetti MITSUBISHI
in confezioni da 10 pz.
2DD 3,5" L. 1.050
2HD 3,5" L. 1.850

Evasione ordini
nelle 24 ore
Sconti su quantitativi

24 aghi colore
300 cps 360 dpi
L. 765.000

GVP Point autorizzato

Assistenza tecnica per tutti i computer
Consulenza specializzata DTV

Tutti i prezzi si intendono comprensivi di IVA

**Disponibili Dischi
BULK e MARCATI**

Vendo Action Replay, ATonce e drive esterno per Amiga. Aldo Franco - Tel. 0984/511453 (ore 10-16).

Varie

Amiga Music Club Italy cerca contatti in tutta Italia. Offriamo software, hardware e manuali ultime novità. Per informazioni: c/o Emanuele Cattaneo - Viale Rimembranze, 18 - 20059 Vimercate (MI) - Tel. 039/6081284.

Appassionato grafica 3D Amiga, con tanti manuali in italiano e soft contatta altri appassionati (anche novizi) per scambi e simpatiche collaborazioni (no approfittatori, no lucro). Fulvio Albrizio - Via Flumendosa, 10 - 20132 Milano - Tel. 02/2562049.

Vendo per Amiga i seguenti corsi in italiano: Imagine v. 2.0, AdPro, Can Do, Amos, Lattice C, Modula II, Cinemorph, Bars e Pipes Pro, Superjam, Octamed, Spectracolor, 3D Professional, Sculpt Animate 4D e tanti altri. No lucro. Per informazioni: Michele - Tel. 02/9960597.

Utilizzo il Commodore 64 e cerco persone che usano lo stesso computer e cerchino contatti per scambio software (giochi e soprattutto utility). Moschetti Stefano - Via San Martino, 4 - 01100 Viterbo (VT).

3000+ Amiga BBS 24h/24h, la prima BBS

italiana dedicata alla grafica. Aree files tutorial in italiano con decine di lezioni e programmi Arexx per: Imagine, Ad Pro, Morphplus, Image FX, Real 3D, 3D Pro, ecc. 3000+ Amiga BBS - Tel. 0544/451764 - 1200 ... 16800 Baud HST.

Compositore di moduli pro/noisetracker cerca acquirenti. Converte su richiesta qualsiasi musica da discoteca (disponibili: No-limits, Transformer II, Open your mind, e molte altre). Inviare Lire 5.000 + spese postali (Lire 2.000) per demo-disk a: Davide Filidei - Via Case Bianche, 2 - 56030 Calcinaia (PI).

Cerco Amanti di grafica al computer per scambio idee e collaborazioni su programmi. Tel. 039/304700 - Michele (ore 18.00-21.00).

AmigaLink Mid-Italy BBS tutto il software FW e SW per Amiga con particolare attenzione alla grafica 2D-3D. Accesso gratuito e D/L illuminato. Un club di amici per tutti gli hobbisti Amiga. Attiva 24h - Tel. 0564/415697 - 411380 - 2400 + 14400.

Light & Magic Club offre agli appassionati del DTV su Amiga il meglio della propria produzione: Textures, Brush, Clip Art, animazioni e titoli per cerimonie 2D e 3D, immagini digitalizzate, Backgrounds, Fonts a colori. Il tutto anche a 24 bit, Ham 8, etc. Tel. 0337/972908 - Michele.

Cerco appassionati geometria frattale (soprattutto paesaggi) per scambio di conoscenze. Risposta assicurata. Cerco distributori di software PD in Torino e provincia. Inviare liste e prezzi senza impegno da parte mia. Marco Deferre - Via

Bonfante, 7 - 10137 Torino.

Facciamo musica techno/rave a livello professionale. Cerchiamo e scambiamo moduli (Protracker e Octamed pro. 3.0) e samples (non meno di 25 MHz) rigorosamente techno/rave/house. Possediamo più di 800 samples. Per informazioni: Tel. 0436/890406 - 99086 - Mattia.

Commodore Club

Circolo Idea CDTV. In sede è a disposizione dei soci un CDTV e sono disponibili (anche a casa) programmi compact CDTV e per Amiga 500, ci sono giochi da tavolo e servizio bar. Per scambi o una serata diversa, scrivete o venite al circolo Idea - Corso Fiume, 40 - 10046 Poirino (TO). Venerdì e sabato orario 21.00 - 00.30 (suonare il campanello illuminato).

Amiga Club-Cremona invita tutti i possessori di Amiga a contattare il club per ricevere gratuitamente la nostra documentazione. Scrivere a: Amiga Club c/o Paolo Dilda - Via Giuseppina, 178 - 26048 Sospiro (CR).

Amiga User's Club cerca soci nella zona di Ferrara. L'iscrizione è gratuita e non comporta alcun impegno. Si accettano utenti di ogni tipo di sistema Amiga (pref. S.O. 2.0 e superiori). Telefonare dalle ore 14.00 alle ore 17.00 da lunedì a giovedì allo: 0532/204765.

INDICE DEGLI INSERZIONISTI

Inserzionista	Pag.
Abacus	5
A.P.S.	22
Commodore Italiana	7
Computer Service	93
Data Office	93
DB Line	63, 91
DeskTop Video	88
Flapperia	2
Hardital	89
Hi-Fi Club	91
IHT Gruppo Editoriale	II, III, IV, 1
Nawel	8, 23
Nex	49, 50
Rainbow Computing	55
R.S.	45
Studio Birplane	63
Supergames	17

Direzione vendite spazi pubblicitari:

IHT Gruppo Editoriale - Commodore Gazette
Via Monte Napoleone, 9 - 20121 Milano
Tel. 02/794181 - 799492 - 76022612
Telex 334261 IHT I - Telefax 02/784021

Questo indice è da considerarsi come un servizio aggiuntivo. L'Editore non si assume alcuna responsabilità per eventuali errori e/o omissioni. Indirizzare eventuali lamentele riguardanti gli inserzionisti a:

Commodore Gazette - Uffici Pubblicitari
Via Monte Napoleone, 9 - 20121 Milano

Nessuna responsabilità viene altresì assunta dalla Commodore Gazette per eventuali problemi di qualsiasi natura con gli inserzionisti. La responsabilità di quanto pubblicato negli spazi pubblicitari è esclusivamente del committente. Anche se per motivi di spazio non sono stati inseriti in questo indice, anche per gli inserzionisti presenti nella rubrica di inserzioni a pagamento "Pagine Gialle" valgono le medesime condizioni che regolano i rapporti con gli inserzionisti inseriti in questo indice.

COME DIGITARE I LISTATI DI COMMODORE GAZETTE

I listati per C-64/128 contengono una particolare simbologia. Tutti i caratteri grafici e quelli di controllo sono stati tradotti in combinazioni di tasti facilmente comprensibili. Sono le istruzioni tra parentesi graffe. Per esempio, {SHIFT L} indica che si deve tener premuto il tasto shift e premere una volta il tasto L. Ovviamente, non bisogna digitare le parentesi; quello che apparirà sullo schermo saranno simboli grafici. Altri esempi:

{20 SPAZI} premere la barra spaziatrice 20 volte.

{SHIFT CLR} tenere premuto il tasto shift e premere una volta il tasto clr-home.

{2 CRSR } premere cursore-giù due volte.

{CTRL 1} tenere premuto il tasto control e premere il tasto 1.

{COMD T} tenere premuto il tasto col logo Commodore e premere T.

{CRSR ←} premere cursore-sinistra una sola volta.

{SHIFT A} tenere premuto il tasto shift e premere il tasto A.

Gli altri tasti che non danno origine a caratteri particolari (come ↑, ↓, @) sono invece presentati normalmente.

Manoscritti: le collaborazioni dei lettori - manoscritti, disegni e/o fotografie - sono benvenute e verranno valutate in vista di una possibile pubblicazione. Commodore Gazette non si assume comunque responsabilità per perdite o danni al materiale. Si prega di allegare una busta affrancata e indirizzata per ogni articolo. Il pagamento per materiale non richiesto viene effettuato solo in seguito all'accettazione da parte della redazione. I contributi editoriali (di qualunque forma) non si restituiscono. Tutta la corrispondenza editoriale, richieste di annunci, problemi di sottoscrizione abbonamenti, di diffusione e con gli inserzionisti, deve essere indirizzata a: Commodore Gazette - Uffici Editoriali - Via Monte Napoleone, 9 - 20121 Milano. **Commodore Gazette** è un periodico indipendente non connesso in alcun modo con la Commodore Business Machines e con tutte le sue sussidiarie e affiliate, compresa la Commodore Italiana S.p.A. Commodore Gazette viene pubblicata dalla IHT Gruppo Editoriale, Via Monte Napoleone 9, 20121 Milano. Nessuna parte di questa pubblicazione può essere in alcun modo riprodotta senza il permesso scritto dell'editore. La redazione si adopera per fornire la massima accuratezza negli articoli e nei listati pubblicati. Commodore Gazette non si assume responsabilità per eventuali danni dovuti a errori ad omissioni.

IL PROSSIMO NUMERO SARÀ IN EDICOLA IL 20 GIUGNO

SERVIZIO LETTORI

Questa scheda è valida fino al 20 giugno 1993

A. Come giudica questo numero di Commodore Gazette?

- ☐ 1. Ottimo
- ☐ 2. Molto buono
- ☐ 3. Buono
- ☐ 4. Discreto
- ☐ 5. Sufficiente
- ☐ 6. Mediocre
- ☐ 7. Insufficiente

B. Quale(i) articolo(i) di questo numero ha apprezzato maggiormente?

C. Quale(i) articolo(i) di questo numero giudica peggior(i)?

D. Quali argomenti dovrebbero essere trattati nei prossimi numeri di Commodore Gazette?

E. Con quale aggettivo descriverebbe Commodore Gazette?

F. Quante persone leggono la sua copia di Commodore Gazette?

- ☐ 1. Una
- ☐ 2. Due
- ☐ 3. Tre
- ☐ 4. Quattro o più

G. Ha dei suggerimenti?

H. Quale(i) computer utilizza?

- ☐ 1. C-64
- ☐ 2. C-128/C-128D
- ☐ 3. Amiga 500
- ☐ 4. Amiga 600
- ☐ 5. Amiga 1200
- ☐ 6. Amiga 2000
- ☐ 7. Amiga 3000
- ☐ 8. Amiga 4000
- ☐ 9. CDTV

☐ 10. Altro (specificare) _____

I. Quale(i) computer intende acquistare nel futuro?

- ☐ 1. C-64
- ☐ 2. Amiga 600
- ☐ 3. Amiga 1200
- ☐ 4. Amiga 2000
- ☐ 5. Amiga 3000
- ☐ 6. Amiga 4000
- ☐ 7. CDTV
- ☐ 8. Altro (specificare) _____

L. È un acquirente dei libri della IHT? Se sì, come li giudica?

M. Ha mai visto la trasmissione Informatica VideoMagazine? Se sì, come la giudica?

N. Indichi in ordine di classifica le riviste d'informatica che giudica migliori

1. _____
2. _____
3. _____

O. Indichi quali sono i suoi maggiori interessi

- ☐ 1. Videoregistrazione
- ☐ 2. Hi-Fi
- ☐ 3. Strumenti musicali
- ☐ 4. Fotografia
- ☐ 5. Automobili
- ☐ 6. Altro (specificare) _____

P. Quali periferiche intende acquistare nei prossimi sei mesi?

Q. Quanto intende spendere in software e hardware nei prossimi sei mesi?

Nome e cognome _____

Indirizzo _____

Città _____

Prov. _____

C.a.p. _____

Età _____

Professione _____

COMMODORE
GAZETTE

maggio 1993

SCHEDA ORDINAZIONE LIBRI E VIDEO

Con il presente tagliando desidero ordinare il(l) seguente(e) libro(i):

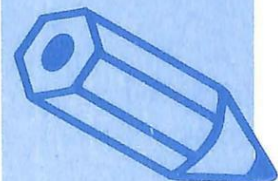
Collana Informatica

- | | | |
|--|---|-----------|
| ☐ L'Amiga | (Michael Boom) | L. 60.000 |
| ☐ Il Manuale dell'AmigaDOS | (Commodore-Amiga) | L. 60.000 |
| ☐ Programmare l'Amiga Vol. I | (Eugene P. Mortimore) | L. 80.000 |
| ☐ Programmare l'Amiga Vol. II | (Eugene P. Mortimore) | L. 70.000 |
| ☐ Il Manuale dell'hardware dell'Amiga | (Commodore-Amiga) | L. 76.000 |
| ☐ Guida ufficiale alla programmazione di GEOS | (Berkeley Software) | L. 64.000 |
| ☐ Flight Simulator Co-pilot | (Charles Gulick) | L. 30.000 |
| ☐ Volare con Flight Simulator | (Charles Gulick) | L. 45.000 |
| ☐ Le mille luci di Hollywood | (David Chell) | L. 42.000 |
| ☐ Inventori del nostro tempo | (Kenneth A. Brown) | L. 42.000 |
| ☐ Computer in guerra: funzioneranno? | (David Bellin e Gary Chapman) | L. 39.900 |
| ☐ La sfida della crescita | (G. Ray Funkhouser e Robert R. Rothberg) | L. 39.900 |
| ☐ La macchina e la Mente | (George Johnson) | L. 42.000 |
| ☐ I Creatori del Domani | (Grant Fjermedal) | L. 39.900 |
| ☐ L'Universo del Giovedì | (Marcia Bartusiak) | L. 39.900 |
| ☐ Frontiere Invisibili | (Stephen Hall) | L. 54.000 |
| ☐ Computerarte, computergrafica e animazioni vol. I | (IHT Video) | L. 39.900 |
| ☐ Computerarte, computergrafica e animazioni vol. II | (IHT Video) | L. 39.900 |

Collana Cinema Collana Tempus

Videocassette

Pagherò in contrassegno al postino la somma di L. + spese postali (L. 8.000 per volume)



Nome e cognome _____

Indirizzo _____

Città _____

Prov. _____

C.a.p. _____

Tel. _____

Firma _____

COMMODORE
GAZETTE

maggio 1993

13-5

☐ Desidero inserire gratuitamente un mio annuncio nella rubrica CLASSIFIED (solo per i privati e per gli annunci non a scopo di lucro).

Attenzione: perché un annuncio venga accettato è necessario che sia stato compilato anche il questionario presente sull'altro lato di questo tagliando. Non si accettano fotocopie.

TESTO: _____

Inserire all'interno di una busta affrancata e spedire a:

**Commodore Gazette
Servizio Lettori
Via Monte Napoleone, 9
20121 Milano**



Inserire all'interno di una busta affrancata e spedire a:

**Commodore Gazette
Servizio Lettori
Via Monte Napoleone, 9
20121 Milano**

Oppure inviare via fax allo 02/784021

ABBONARSI A COMMODORE GAZETTE GRATIS!



Grazie a questa straordinaria offerta, un abbonamento a Commodore Gazette può essere praticamente gratuito. Infatti, chi si abbona spedendo il tagliando di questa pagina riceve in omaggio un libro di un valore che può essere anche superiore al costo dell'abbonamento.

ECCO UN ESEMPIO

Abbonamento
a 11 numeri: 88.000

Sconto dell'offerta: .. -19.000

69.000

1 libro
(Programmare
l'Amiga vol. II) -70.000

Totale -1000!!!

I VANTAGGI DELL'ABBONAMENTO:

-  **OLTRE IL 20% DI SCONTO
SUL PREZZO DI COPERTINA**
-  **UN LIBRO IN REGALO
A VOSTRA SCELTA**
-  **SICUREZZA DI NON PERDERE
NEANCHE UN NUMERO**
-  **COMODITÀ DI RICEVERE
LA RIVISTA A CASA**
-  **PREZZO BLOCCATO
IN CASO DI AUMENTI**

Sì, mi abbono a 11 numeri della rivista Commodore Gazette a partire dal numero ____
Usufruirò così dello sconto di oltre il 20% sul prezzo di copertina e riceverò un libro gratuitamente.

Nome e Cognome _____

Indirizzo _____

Città _____ C.A.P. _____

Allego assegno bancario, postale, circolare, o fotocopia della ricevuta di un vaglia postale, intestato alla IHT Gruppo Editoriale per l'importo di lire 69.000.

Riceverò in omaggio a casa mia il seguente libro (indicare con una crocetta il libro scelto):

- | | |
|--|---|
| ☐ L'Amiga | ☐ Inventori del nostro tempo |
| ☐ Il Manuale dell'AmigaDOS | ☐ Computer in guerra: funzioneranno? |
| ☐ Programmare l'Amiga Vol. II | ☐ La sfida della crescita |
| ☐ Guida ufficiale alla programmazione di GEOS | ☐ La Macchina e la Mente |
| ☐ Flight Simulator Co-pilot | ☐ I Creatori del Domani |
| ☐ Volare con Flight Simulator | ☐ L'Universo del Giovedì |
| ☐ Le mille luci di Hollywood | ☐ Frontiere Invisibili |

Firma _____

SCRIVERE IN STAMPATELLO IN MODO CHIARO E LEGGIBILE

Ritagliare e spedire a: IHT Gruppo Editoriale - Via Monte Napoleone, 9 - 20121 Milano

IN TV È NATA



INFORMATICA VIDEOMAGAZINE

Il primo settimanale televisivo dedicato al mondo dei computer

Ogni settimana su ODEON TV,
alle 23.15 del mercoledì
e in replica alle 14.45
del sabato, c'è un appuntamento
imperdibile: **INFORMATICA
VIDEOMAGAZINE (IVM)**,
una trasmissione televisiva
dedicata al mondo dei computer.



Ecco alcune delle rubriche fisse di IVM:

- **Computergrafica e computerarte:** il meglio della *computer animation* realizzata a livello mondiale con interviste ai suoi creatori.
- **TG Informatica:** il primo TG interamente dedicato all'*information technology*.
- **Videoprove hardware e software:** il mezzo televisivo diventa il vostro "occhio" per esplorare ciò che offre il mercato.
- **E ancora:** dibattiti, interviste, inchieste, servizi speciali...

**VI ASPETTIAMO OGNI MERCOLEDÌ ALLE 23.15
E IN REPLICA AL SABATO ALLE 14.45 SU**

ODEON

Per informazioni:
Informatica VideoMagazine
Via Monte Napoleone, 9
20121 Milano
Ufficio pubblicità: ☎ (02) 794181
Redazione: ☎ (02) 76022612

IVM è una trasmissione prodotta
da IHT Gruppo Editoriale
e Datanord Multimedia.